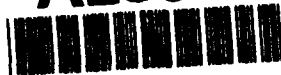


AD-A280 219



UNIVERSITY OF CALIFORNIA, SAN DIEGO

BERKELEY • DAVIS • IRVINE • LOS ANGELES • RIVERSIDE • SAN DIEGO • SAN FRANCISCO



SANTA BARBARA • SANTA CRUZ

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

9500 GILMAN DRIVE
LA JOLLA, CALIFORNIA, 92093-0407

April 15, 1994

Re: Semi-annual report on ONR grant no: N00014-91-J-1017

Title: Fault Tolerance in Opto-electronic Computing

Principle Investigator: Professor Ting-Ting Y. Lin

Addressees:

Scientific Officer, Dr. Clifford G. Lau

Administrative Grants Officer

Director, Naval Research Laboratory

Defense Technical Information Center



Dear Sirs,

This letter report, for the period of 1 October 1993 through 1 April 1994, describes the activities supported by the grant in three sections: research progress, student profile, PI's activity.

1. Research Progress

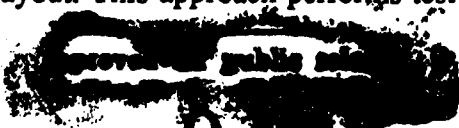
We have been quite successful in getting our work known to the community in the past few months. Specifically, in the several conferences we participated, people have shown great interests.

A. Fault Modeling and Testing of Complex Systems

This project has gotten some recognition in both fault modeling and testing of optical interconnects in a opto-electronic system. In the first part, we want to develop a new methodology for performance analysis of opto-electronic systems at a higher level. Our approach is to first identify possible failures in such interconnect implementations, and then extract information from the physical configuration and relate to the system performance parameters. In this way, system-level performance degradation can be estimated to construct design constraint for physical systems. One analysis on link failures in free-space optical interconnects is published in the Proceedings of the SPIE 94, January conference (Exhibit I).

In the second part on testing, we proposed an architecture which integrates the concept of concurrency and distributed test pattern generation for testing complex circuits on a planer layout. This approach performs test pattern generation and response analysis concurrently,

4928
94-17923



94 6 10 070



DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

9500 GILMAN DRIVE
LA JOLLA, CALIFORNIA, 92093-0407

thus minimizing testing time for the overall system. Our approach fully utilizes the fundamentals of BIST circuitry and conducts test pattern generation and output analysis concurrently. The result is published in IEEE Design and Test of Computers, vol. 10, no.4, pp38-51 (Exhibit II). Circuits are partitioned into segments which can be tested in parallel in testing time bounded by 2^n clock cycles, where n is the maximum no. of inputs for the biggest cluster. The result of this study will appear in 1994 Custom Integrated Circuits Conference (Exhibit III). Further hardware reduction can be reduced by binding test circuitry to functional units through logic synthesis and optimization algorithms. We are currently evaluating the impact of test pattern generation issue for the proposed architecture, and the result has been accepted to the 1994 International Conference of ASIC. Simulations are also underway on various benchmark circuit to better observe its overall performance on realistic systems.

B.. Performance Evaluation of Fault Tolerant Optical Multi-stage Interconnection Network

This project helps to set up an evaluation scheme for complex degradable system. We think the uncertainty of optical interconnects supporting fault tolerance can easily be modeled in this framework. An event-driven simulator has been developed to study the corresponding measures as the replicated or dilated banyans degrade under component failure. A fault-injector is used to inject faults under some given distribution and to degrade the network accordingly. In the multipath MINs, different components are given different MTTF for a 4-replicated banyan of size 64×64 implemented with 2×2 switches. In practice, this can reflect a 64 processor 64 module shared memory multiprocessor environment (one of a moderate size), with exactly four distinct paths between any PE-MM pair. A set of design parameters needs be identified at the system level (as opposed to the device level), that can be used to tune up the system behavior to

reflect upon the possible effects of technology. The design parameters are:

- topology
- level of redundancy
- component lifetime
- switching and timing
- source behavior

A detailed discussion on the choice for optical network based on these design parameters is in the book chapter submitted for Kluwer publication. We have some result on a general N component system which has been submitted to 1994 SPDP symposium (Exhibit IV). More experiments designed to give us more insight to the degradation process include a

UNIVERSITY OF CALIFORNIA, SAN DIEGO

BERKELEY • DAVIS • IRVINE • LOS ANGELES • RIVERSIDE • SAN DIEGO • SAN FRANCISCO



SANTA BARBARA • SANTA CRUZ

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

9500 GILMAN DRIVE
LA JOLLA, CALIFORNIA, 92093-0407

convergence test to validate simulation results with a number of fault patterns injected and some time of running time; two other experiments designed to observe the effect of submission rate to mean reward before failure (MRBF) and the mission time. Results on replicated network has been encouraging, we are working on getting results for the dilated network to be published in a journal submission.

2. Student Profile

Amiya Bhattacharya is working on the simulator for performability evaluation. Huoy-Yu Liou continues to develop the framework for the test architecture. She will be taking her research exam early summer. The two new graduate students (De-Yu Kao, and Li-Cheng Tai) are still taking some courses. However, De-Yu passed the comprehensive exam this Spring, which means he will devote more time to research from this point on. His immediate project is to design the controller for the testing architecture.

3. PI's Activity

Other than working with the students on the above mentioned projects, I have been involved in the ASIC and Testing communities. I presented our paper on "Reconfiguration of Fault-Tolerant Linear Processor Arrays" at the International Conference on Parallel and Distributed Systems (ICPADS'93), December 1993. Presented the "Analysis of Link Failures in Free-Space Optical Interconnects" at SPIE Optoelectronic Interconnects Conference, January 1994. I have been invited to the IEEE BIST/DFT workshop to be held later this month (April 19-22) to present our work on the pipelined testing architecture. I also have received invitation to present my work at the MCM Test Workshop this September. I have also been elected to the program committees for the Seventh Annual IEEE International ASIC Conference, the 1994 Symposium on Parallel and Distributed Systems, and the Fourth International Conference on CAD & CG, 1995.

The above detailed my recent research activities. If there are issues omitted or not clear, please do not hesitate to call me. Thank you.

Best regards,


Ting-Ting Y. Lin
(619) 534-4738

Unannounced Justification <input checked="checked" type="checkbox"/>	
By _____ Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

To appear in SPIE Optical Interconnect Conference, January 1994

Analysis on Link Failures in Free-Space Optical Interconnects

Huoy-Yu Liou and Ting-Ting Lin

*Department of Electrical and Computer Engineering,
University of California at San Diego,
La Jolla, California 92093
liou@ece.ucsd.edu, lin@ece.ucsd.edu
(619) 534-4738
(619)534-2486 (fax)*

ABSTRACT

Free space optical interconnect has provided a promising solution to the effective signal links of the increasing density and complexity in very-large-scale/large-scale integrated circuits. It is getting less affordable if such a system fails just for one tiny physical defect. Our analysis on the potential optical-electrical link failure provides guidelines for future testing and reliable system design. The study of fault models starts by exploring the underlying physical mal-functioning of the opto-electrical components, and their impacts on the assembled systems. We map the physical defects of opto-electronic devices into their corresponding logic-level representation for higher level design consideration. This mapping is chosen for its compatibility and practicability with digital electronic system designs where automated design tools can expedite the design optimization and verification process.

1. INTRODUCTION

As the integrated circuits scale down, switching speed and the complexity go up, the demand for high density, larger bandwidth and reconfigurable interconnects becomes crucial for integrated circuits with better performance and higher yield. The free-space optical link in place of conventional metal wiring in semiconductor designs has been under research for over a decade [1]. Research works strive to show that free-space optical interconnect provides one of the best solutions for beyond Gb/s performance ULSI ICs and systems [2][3]. The reasoning is based on the compatibility between semiconductor and optical processing technologies and the potential competitive performance of optical interconnection over metal wiring [4][5]. Therefore, it is essential that physical failures in these optical-electronic hybrid design schemes are understood.

In this paper, we first review current free-space optical interconnect schemes, followed by the proposal of a fault model for opto-electronic hybrid systems. This is achieved by mapping the physical failures of opto-electronic systems into its preliminary logical representation in a comprehensive manner. A short introduction to the existing fault models for the electronic world is provided for quick reference. Finally we take two sample systems for the demonstration of the mapping. At this moment, we will focus on modeling of time-invariant permanent faults where physical failures of components result in permanent system failure.

2. CURRENT IMPLEMENTATIONS OF FREE-SPACE OPTICAL INTERCONNECT

Current implementations of free-space optical interconnects can be categorized into two major families [5]. The first one is characterized as the "modulator" scheme since it adopts external (off-processor or off-chip) light sources and spatial/planar light modulators as the control of the optical paths. One example is the optical interconnect scheme using the SEED devices [6][7][8]. The second family is called the "light source" scheme, which relies on micro-lasers [19] and photo-conductive media/wave guides for one-to-one or one-to-many connections. Some prototypes utilizing microlaser array and planar optics have been proposed [9][10].

In the "modulator" scheme, the photodiodes are built in the communicating processors/devices. The data and/or the control signals are provided by external light sources such as the power-supply lasers. The outputs from each processing element are read out using the read-out laser beam. Interconnection patterns are generated either from a pre-fabricated hologram or a spatial-varying array of holograms. The photo-refractive indices of these holograms are subject to changes depending on the

materials. An applied electro-magnetic field, e.g., a control voltage or changes of wavelength of reference laser beam(s), can be the pattern selector. Reconfigurability of the interconnects is thus possible on the pattern-selection basis [14]. Figure 1 shows an abstract picture of such scheme. Here we use the intensity representation for the digital optical implementation as described in [1].

For the "light source" scheme, each device/processor utilizes micro-lasers as outputs. The micro-lasers emit lights according to the assigned logical values at the outputs of the processor.

In the photo-refractive implementation, the interconnection media can be either a substrate hologram or simply a waveguide connecting the outputs of one device to the input(s) of the other [11][12]. Interconnection patterns can be modulated also. An applied/built-in spatial-variant electromagnetic field to the waveguide substrate or photorefractive gratings will change the diffracting holograms too [15][12]. Those multiplexed planar holograms can provide dynamic reconfigurable interconnect patterns [15]. Figure 2 shows a symbolic sketch of this scheme also using the intensity representation.

These two schemes not only are the major representatives of current studies but are based on fundamental optics. Therefore, we chose them as the main implementations in our discussion of fault modeling.

3. PRESUMPTIONS AND FAULT MODELS IN THE DIGITAL REPRESENTATION

3.1. Presumptions

A few assumptions should be made to better illustrate the modeling process. It is implicitly conceived that analog optics is not under consideration in this article even though some of our discussion may apply to the analog domain.

First, the mechanical alignment problem between optical components is of major concern in today's application of optics. However, two promising techniques which are based on existing semiconductor technology were developed to circumvent alignment difficulties: the self-aligned solder bumping [17] and silicon micromachined etching [18]. The former suggests a sub-micron alignment tolerance in the planar integration of the optical components [11] and both techniques suggest that all of the micro-lasers and photodiodes are to be put abutted to the photo-refractive medium to obtain the best alignment and area efficiency [13]. Therefore, we assume that the mechanical alignment problem can be solved, and we will focus on other random faults.

Secondly, the specific technology of our concern must be silicon-compatible. This is because the integration of opto-electronic components is important to make themselves competitive with their electrical counterpart on performance consideration [25].

Thirdly, we will concentrate on modeling of the permanent physical faults for now. An analysis of the physical faults in each opto-electronic component will be performed in the following to determine the combined effect of these faults in the digital domain. Soft and transient faults, although more frequent, are system-dependent. Further investigation is needed for temporal system-related failure.

All the interconnect faults are transformed to the corresponding digital faults and are presented to the inputs of the receiving processor(s) or devices. Figure 3 illustrates a logic representation of a opto-electronic system using the logic level representation.

3.2. Logic-Level Fault Models for Permanent Failures in Electronic Digital Systems

Before we investigate the possible physical failures of the opto-electronic components, a quick review of the popular fault models in the electronic world is provided. It will benefit the large-scale opto-electronic hybrid designs if we can adopt those techniques into our optical interconnect scheme by a successful mapping from the physical failure of such systems to the corresponding digital representation.

Two major classes of logic-level fault models of the permanent physical failures in electronics, stuck-at and bridging [26]. The stuck-at model simply describes the logical values of the input/output of a gate/block is tied to 0 or 1. Whereas the bridging faults represent the cases of two adjacent wires being physically shorted to one signal line thus issuing one logic value. Those two models gain their popularity because they represent the majority of the hardware failure modes in the LSI/VLSI technology [16]. And testing techniques based on these models can provide satisfactory fault coverage as well as efficient testing time. However in CMOS technology, another fault model is the stuck-open which represents an open connection in the circuit. This fault can transfer a combinational circuit into a sequential machine which still can be simulated on the gate/logic level [26].

Fault simulation and testing techniques rely on effective fault models. A simple logic-level fault model can help speeding up the testing process through the automated fault simulation and test pattern generation. Only by test automation can we

move onto a more cost-effective design cycle of large-scale circuits without sacrificing quality.

3.3. Physical Defects in Optical Components

An outline of possible physical faults of both active and passive devices is given below. In particular, physical defects in the fabrication process of opto-electronic devices are investigated. Their impacts on the digital domain are discussed individually.

Active Devices

- * Processing defects of micro-lasers can shift their resonating mode which may reduce the illuminating intensity of the photodiode [20]. Low intensity as a result, may not trigger the threshold current in order to have a logic true output. This can be viewed as a stuck-at-0 fault at the input of the succeeding processor. On the other hand, a constant high electrical wire/contact will activate the resonator which can be viewed as a stuck-at-1 fault.
- * Abnormal impurity concentration/diffusion in modulators and fan-in/fan-out generators will give too weak (below logic level low threshold) or too strong (above logic level high threshold) light beams under normal operation which appears as the stuck-at faults to the inputs of the succeeding processor(s).
- * Physical defects in photodiodes will result in stuck-at faults. This can be explained by the fact that unexpected impurity concentration or crystalline imperfection will adjust the energy barrier of the proposed Schottky diodes [24] or the built-in voltage which takes part in off-setting the total photo-induced current of the p-i-n diodes [21].
- * Defects in the micro-lasers/photodetectors can introduce fluctuation of the recovery time which gives rise to delay faults. It has been derived that the turn-on time of the semiconductor laser/diode is proportional to the carrier life time which is relevant to the impurity concentration [21].

Passive Devices

- * The defects of the micro lenses/gratings or diffractive optical elements [13] will give less reflected intensity, shown in Figure 4a as the stuck-at-low faults. Whereas in Fig. 4b, too much reflected light results in stuck-at-high or bridging faults where two or more signals are collected to one photodetector by a faulty array of grating. It is possible when the array of diffracting components fail to link between two processing elements, such that no signal is present at the receiving end. We call this "link-open" fault, and designate an unknown state in our future simulation.
- * The overwriting problem of the bulk and planar holograms during read out [12] can contribute to the permanent stuck or bridging faults. Transient failure can be expected with the fluctuation of temperature, voltage or laser power.
- * Potential crosstalk, light scattering and noise in the optical fiber/waveguide [22][23] can result in transient or bridging faults. (However, this can be avoided by introducing new Design Rules between the two light paths.)

The above list is an outline of the most common physical faults in current implementation. We will illustrate the models using the following applications.

3.4. Engineering/Performance Parameters Representing Physical Failures

There are several common engineering/performance parameters in designing the free-space optical interconnects. Here we try to link the parameters with digital fault models to make a quantitative representation of the permanent faults for system design evaluation. This is particularly helpful for optical-electric system designers to conduct system level simulation for automated design planning. Since free-space optical interconnect has to handle massively parallel interconnection, some of the parameters are included for array-type digital interconnection in the system. The discussion is itemized as following:

- * *Misalignment* [27] -- Deviation in incident light wavelength, lenslet focal length, image plan tilt, non-planar light wave curvature, and distance between lenslet and image plan, can cause misalignment. Alignment failure (i.e., non-tolerable deviation of the above physical quantities in designing a system) in the laser array generator can result in bridging/wired-or ("cross-talk"), stuck-at, or link-open faults.

- * *Light Throughput* [28]/*System Power Transmission* [27] -- The light efficiency of a free-space optical interconnect system can be defined as the product of light/power efficiency of all components used for the interconnect, i.e.,

$$\eta_{System} = \prod_{i \in System} \eta_{ith-component}$$

Therefore, the fluctuation of the power efficiency, $\Delta\eta_{System}$, can be derived as,

$$\Delta\eta_{System} = \Delta\left(\prod_{i \in System} \eta_{ith-component}\right) \approx \sum_{i \in System} \left(\prod_{j \neq i} \eta_{jth-component}\right) \times (\Delta\eta_i), \text{ for } \Delta\eta_i < \eta_i < 1 \quad (1)$$

For a given photodetector set, the run-time power fluctuation in the components should be carefully restricted to give a satisfactory tolerance at the detectors. Non-tolerable light/power fluctuation can cause stuck-at or link-open faults.

* **Signal Noise Ratio (SNR) and Bit Error Rate (BER)** [28] – For a symmetric binary channel with Gaussian noise, the BER can be written in terms of the SNR as

$$BER = 0.5 (1 - \text{erf}(SNR/2\sqrt{2})) \quad (2)$$

When a free-space optical interconnect system is modeled by Eq. (2), the upper bound of the SNR of the system is given by the performance requirement, the BER. Once the SNR function (in terms of the physical parameters of the components) of a system is given, the noise margin is set for the system. For example, if $BER < 10^{-12}$, then $SNR > 14$. Using the SNR given by Eqs. (B.1), (B.2), and (B.3) for spacial invariant optical interconnect system in [28], we have

$$SNR = \frac{\eta_1}{N_s \times 2 \times w_s \times \sum_{i \neq 1} \eta_i / A_i} > 14.$$

A relation can thus be derived between η_1 and η_i 's in terms of N_s , the number of light sources, w_s , the signal area, and A_i 's, the area covered by higher order phase from the diffractive lenses,

$$\frac{\eta_1}{28 \times N_s} > \sum_{i \neq 1} \left(\frac{w_s}{A_i} \right) \times \eta_i. \text{ Any combination of } w_s \text{ and } A_i \text{'s breaking this inequality will fail to meet the system performance requirement specified by the BER.}$$

4. APPLICATIONS OF THE PROPOSED FAULT MODELS

4.1. Modulator Scheme

In the SEED Based Optical Interconnect [29] example, the SEED devices are used as the output pads for inter-chip/board communication. The laser diodes provide the light source for the input and output of the SEED modulator. And the lenses, gratings, polarization beam splitters and mirrors are used to adjust the optical path way. The output light from a SEED is reflected by the polarization beam splitter and mirror(s) to the Si photodiode array, where the Si CMOS photodiodes are chosen as the input pads for each processing module. (Fig. 9 of [6] shows this interconnect scheme.). SEEDs and photodiodes are fabricated on the same substrate as that of the processing elements. The laser source provides the light for the input and output of a SEED modulator. The output light from a SEED gets reflected by the polarization beam splitter and mirror(s) to the Si photodiode array.

In the preceding setup, permanent defects can be found responsible to the misalignment problem. Misalignment will result in stuck-at faults in the modulators, photodiodes, imperfect beam splitter, gratings, and mirrors. Faulty mirrors and beam splitters could result in bridging and link-open faults. The overwritten problem for holograms can be included if the beam splitting pattern and/or the Damman grating are recorded as holograms.

Unfortunately, the faults are not exclusive to each other. The worst case will be a stucked modulator output distributing its faulty signal to many photodiodes which causes several stuck-at faults at the input of the next module(s). This is shown in both figures 5a and 5b where possible stuck-at and bridging faults can be found at the inputs of the processors in this scheme.

4.2. Light Source Scheme

In the "light source" scheme, the VLSI-integrated transducer [30] is used as our example system. In this system, the outputs of the processing elements are transformed into light signals by the array of micro-lasers and semiconductive photodiodes are used as the inputs. The micro-lasers and photodiodes can be fabricated with the electronic circuitry on the same substrate. Micro lenses, mirrors, gratings and/or beam splitters are fabricated on one side of the glass substrate.

The InP substrate with reflective surfaces, mirrors and gratings provide the transmission media and interconnect patterns. Figure 4 in [30] gives an abstract view of this scheme.

Those problems from micro-lasers, photodiodes, etc. can induce stuck-at faults. Since bridging and link-open faults are less likely from the lasers and diodes, we can conclude that the bridging and link-open faults are from passive devices. In the worst case, when a micro-laser and a part of the holographic pattern go wrong, a distributed stuck faults at the inputs of the next module(s) results. This is shown in figures 6a and 6b. However, the existence of the bridging or link-open faults can tell us the light guiding medium is faulty.

Through the above discussion, one can conclude that the mapping of physical defects into digital representation for the

free space optical interconnect systems is very applicable. Once the mapping mechanism is established, we can talk about fault-tolerant opto-electronic system design and testing in the digital world more efficiently.

5. CONCLUSION AND FUTURE WORK

We have studied the physical defects in the components used by current implementations of free-space optical interconnect. Extraction of the physical failure to a higher level of representation has been demonstrated both qualitatively and quantitatively. This information abstraction can help system designer perform early-stage trade-off analysis and simulation using automated design tools.

In the hybrid systems which we discussed in this work, it is possible to adopt the existing circuit testing techniques once the mapping is done. As soon as the proposed systems are detailed, people can start thinking about the automated testability of such hybrid systems. Whether those designs for testability are to be implemented by optical components or electronics should be left to more specific discussion for different systems. Furthermore, a higher-level design-for-testability and fault-tolerant design methodology is made possible once the potential failures are identified for a free-space optical interconnect system.

There are a lot of topics this work may induce. Soft faults, time-varying/transient faults, and fault-tolerant system-level design are our immediate goal for further investigation. Moreover, the assumption that optical-to-electrical conversion is perfect and we can be relaxed for detail signal analysis. A proper evaluation methodology is also needed to characterize system performance. People should think about the optical-electrical converters when we get into detail signal analysis.

6. REFERENCES

1. Dror G. Feitelson, "Digital Optical Computers", Optical Computing, The MIT Press, 1990.
2. Joseph W. Goodman et al., "Optical Interconnections for VLSI", Proc. of the IEEE, vol. 72, no. 7, 1984.
3. M. R. Feldman et al., "Comparison between optical and electrical interconnects based on power and speed considerations", Appl. Opt., vol. 27, no. 9, 1989.
4. John D. Crow, "Optical Interconnect Technology for Multiprocessor Networks", Digital Optical Computing, edited by Ravinda Athale, 1990.
5. J. L. Jewell et al., "Nonlinear FP etalon and microlaser devices", Digital Optical Computing, edited by Ravinda Athale, 1990.
6. Micheal E. Prise, "Optical computing using Self-Optic Effect Devices", Digital Optical Computing, edited by Ravinda Athale, 1990.
7. A. Dickinson et al., "Free-space optical interconnection scheme", Appl. Opt., vol. 29, no. 14, 1990.
8. F. B. McCormick et al., "Optical circuitry for free-space interconnections", Appl. Opt., vol. 29, no. 14, 1990.
9. Jeurgan. Jahns et al., "Optical Interconnects Using Microlasers and Planar Optics", Proceedings of 1990 SPIE Conf. in Japan, 1990.
10. H. J. Haumann et al., "Optical bus based on light-guiding-plates", Proc. of SPIE Conf. in Japan, 1990.
11. J. W. Parker, "ESPRIT II OLIVES - A European Collaborative Programme in Optical Interconnections", Proc. of SPIE Conf. in Japan, 1990.
12. Lambertus Hesselink, "Photorefractive Approaches to Optical Interconnects", Digital Optical Computing, edited by Ravinda Athale, 1990.
13. Jeurgan Jahns and A. Huang, "Planar integration of free-space optical components", Appl. Optics., vol. 28, no. 9, 1989.
14. Shudong Wu et al., "Reconfigurable interconnections using photorefractive holograms", Appl. Optics., vol. 29, no. 8, 1990.
15. Freddie Lin et al., "Optical multiplanar VLSI interconnects based on multiplexed waveguide holograms", Appl. Optics., vol. 29, no. 8, 1990.
16. Barry Johnson, "Design and Analysis of Fault Tolerant Digital Systems", Addison-Wesley, 1989.
17. D. J. Pedder, "Flip-chip Solder Bonding for High Speed Computing", Plessey Res. Rev., 1989.
18. K. E. Peterson, "Silicon as a Mechanical Material", Proc. IEEE, vol. 70, no. 5, 1982.
19. A. Scherer et al., "Fabrication of Low Threshold CW Electrically Pumped Surface Emitting Microlasers", IEEE LEOS, 1989.
20. Orazio Svelto, "Principles of Lasers", 2nd. ed., Plenum Press, New York, 1982.
21. S. M. Sze, "Physics of Semiconductor Devices", 2nd. ed., John Wiley and Sons, 1981.
22. L. A. Hornak et al., "Wafer-level Optical Interconnection Network Layout", Optical Interconn. Networks, SPIE, vol. 1281, 1990.
23. George Chiu et al., "Overview of Optical Interconnect Technology", Proc. of Internl. Conf. on Adv. in Interconn. Packag-

ing, SPIE, vol. 1389, 1990.

24. D. L. Crawford et al., "High Speed Photodetectors", Digital Optical Computing II, SPIE, vol. 1215, 1990.
25. D. Miller, "Device requirements for digital optical processing", Digital Optical Computing, edited by Ravinda Athale, 1990.
26. Y. Levendel et al., "Fault Simulation", Fault-Tolerant Computing, edited by D. K. Pradhan, vol. 1, Prentice Hall, 1986.
27. F. B. R. McCormick, "Array Testing and Operation", FET-SEED Workshop, AT&T Training Material, June, 1993.
28. M. Feldman, "Optical Interconnections for VLSI", Chap. 4, Ph.D. Dissertation, 1989.
29. H. S. Hinton, "SEED-Based Switching Systems", FET-SEED Workshop, AT&T Training Material, June, 1993.
30. A. J. Moseley et al., "Eight channel InGaAs/InP quantum well asymmetric Fabry-Perot modulator hybridized with foundry VLSI silicon CMOS circuits", Electronic Lett., vol. 18, no. 17, Aug., 1992, pp. 1658-60.

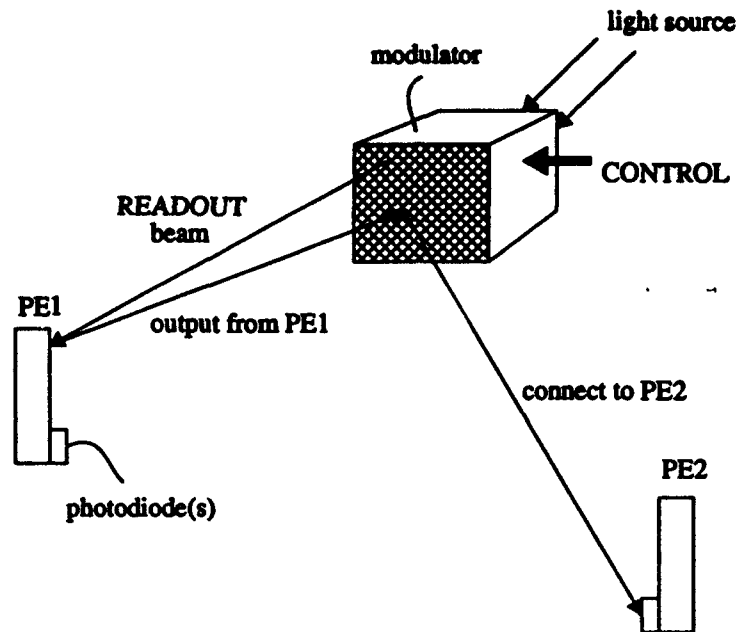


Fig.1 Schematic representation of the "modulator" scheme.
Each processing element (PE) has photodiode(s) for the receiving of optical signal.

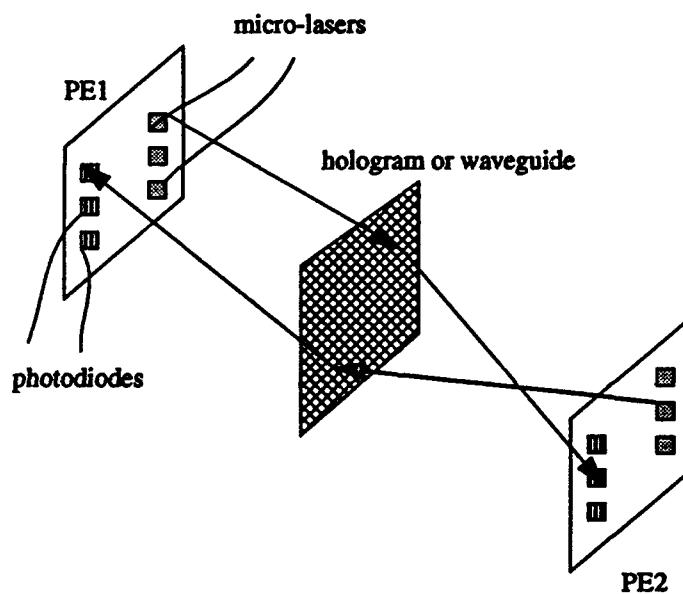


Fig. 2 Schematic representation of "light source" scheme.
Each processing element has both the light sources (micro-lasers) and photodetectors for the data communication

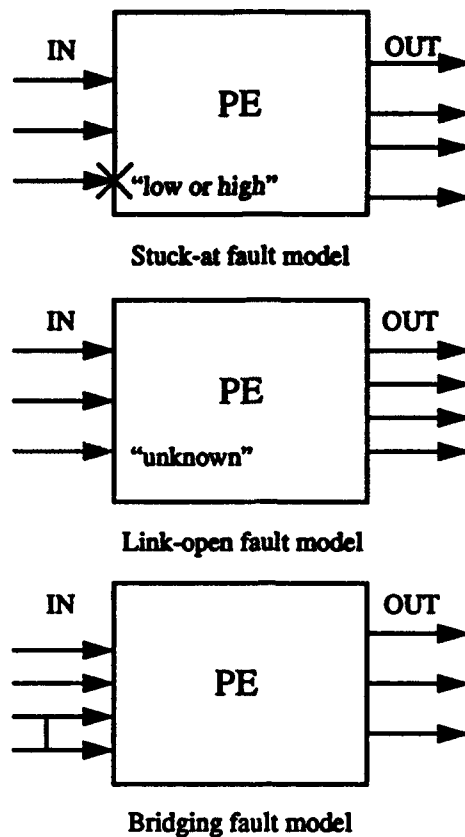


Fig.3 Logical representation of the permanent physical faults from free-space optical interconnect components.

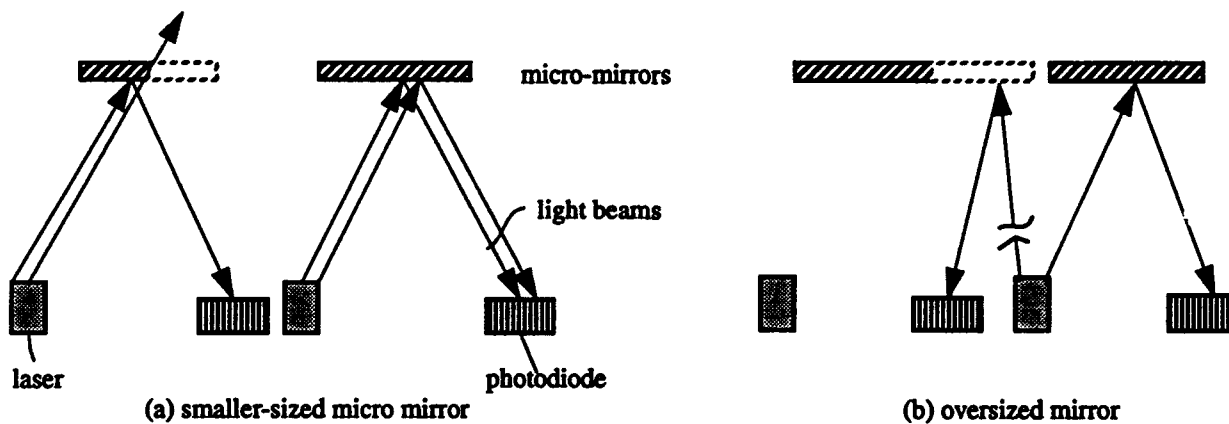
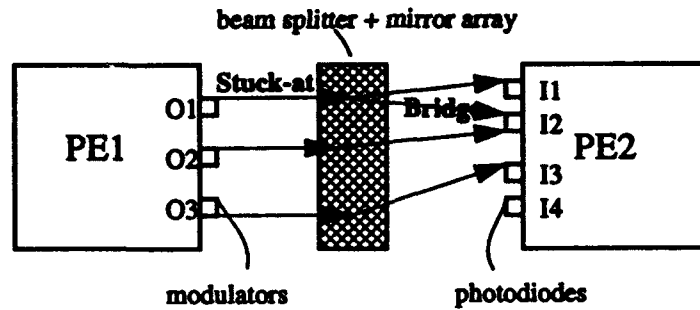
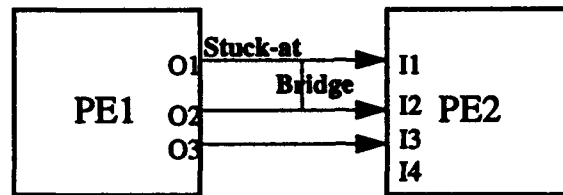


Fig.4 Processing defects for the micro mirrors can introduce two primary physical faults for (a)smaller (b)oversized cases. The latter is not necessarily the nearest neighbor.

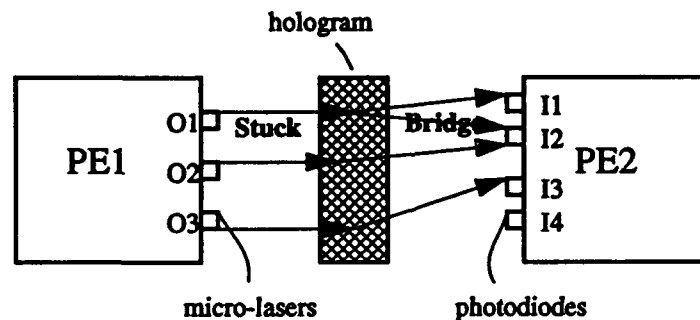


(a) worst-case of a stuck modulator, O1, and a bridging interconnect pattern will cause a faulty input for I1.

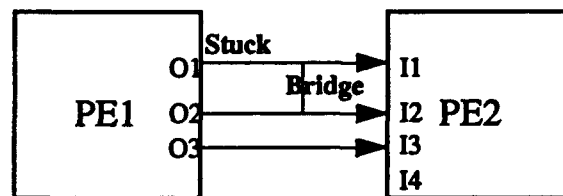


(b) logical representation of the combined failure from (a)

Fig. 5 The worst-case of a combined failure in the "modulator scheme" can distribute the stuck fault to several inputs of the succeeding processing module.



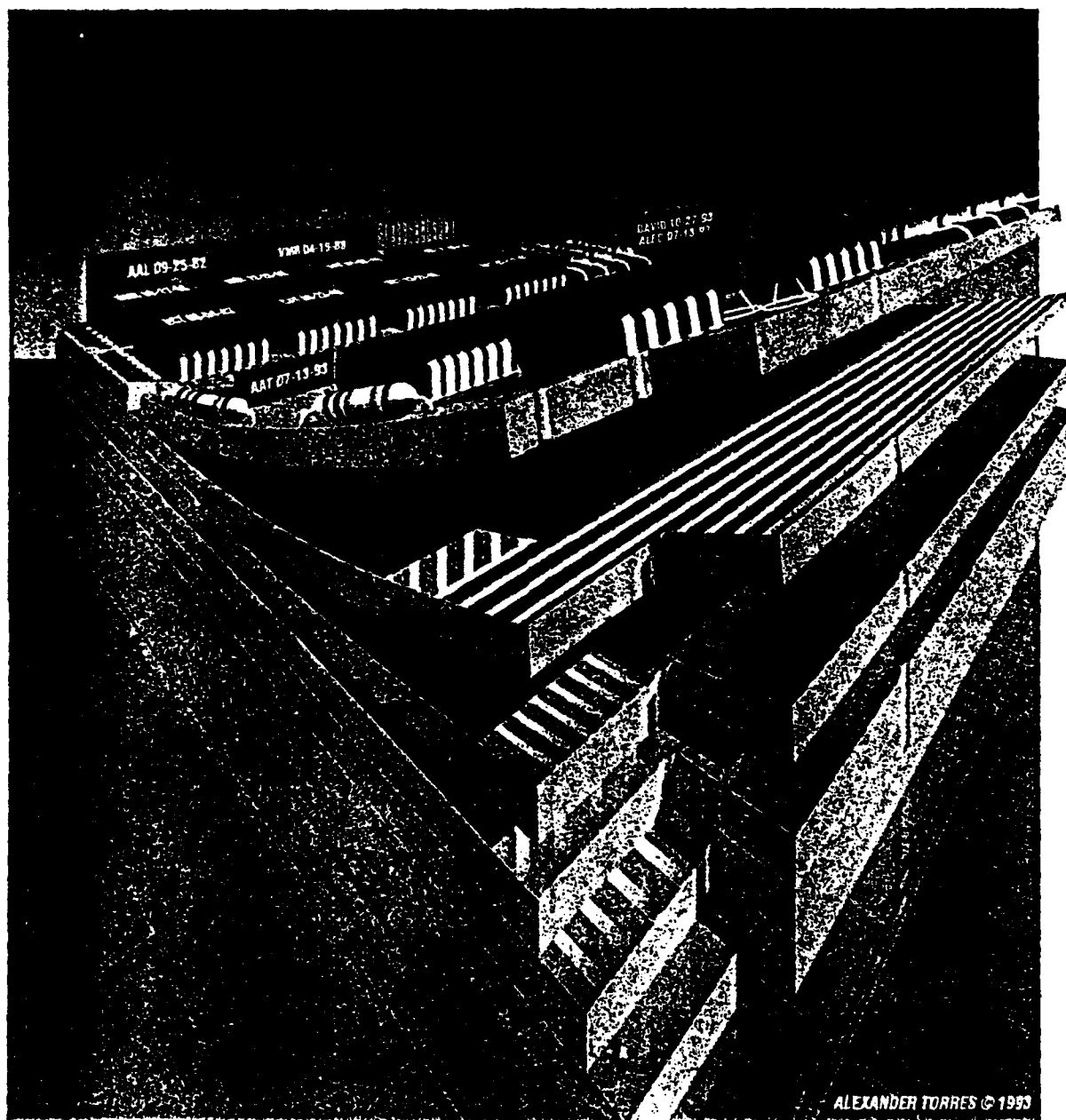
(a) worst-case of a stuck micro-laser, O1, and a bridging interconnect pattern will cause a faulty input for I1.



(b) logical representation of the combined failure from (a)

Fig.6 The worst-case of a combined failure in the "light source scheme" can distribute the stuck fault to several inputs of the succeeding processing module.

Design & Test of Computers



ALEXANDER TORRES © 1993

Multichip Modules

- Hardware/Software Codesign, Part 2
- A Fault-Tolerant Digital Artificial Neuron
- 1993 Annual Index

A New Framework for Designing Built-in Test Multichip Modules with Pipelined Test Strategy

MULTICHIP MODULES interconnect multiple bare dies by means of a stack of conductive and dielectric thin film. As the next generation of electronic packaging, MCMs offer tremendous advantages such as reduced time delays between chips, less electrical noise and cross talk, simplified power distribution, and small size. However, large I/O lead counts and high-density interconnections decrease testing throughput and accelerate testing cost. Traditional hierarchical testing, which involves testing chips individually before assembly and then testing the assembled module to avoid any errors introduced during packaging, includes bed-of-nails fixtures and hand-held diagnostic probes. These methods become impractical and costly with new technologies such as MCMs and surface-mounted devices. Incomplete or unavailable test vectors from chip manufacturers and the internal module's low observability contribute to these problems. Built-in test, where the circuit or system under test includes a small circuit for testing, represents a new approach in testing. Exam-

TING-TING Y. LIN

HUOY-YU LIU

University of California
at San Diego

The authors introduce a novel test strategy, the loop testing architecture, to reduce aliasing probability and testing time for multichip modules. Comparisons to other approaches confirm that LTA provides a new framework for designing effective testable systems.

ples of well-known BIT techniques include the scan design and built-in logic block observer methods.^{1,2}

Scan-design methods involve disconnecting the memory elements and/or the flip-flops from the combinational logics. The overwhelming number of test outputs generated by a relatively large circuit makes the scan method cumbersome. Signature analysis, a popular data-compaction solution, uses a

linear-feedback shift register to receive and modify output data. After a long sequence of test patterns, the residue (or signature) in the shift register of a faulty circuit differs from the signature of a good circuit. Therefore, combining the boundary-scan and built-in self-test techniques provides an alternate method for testing complex circuits at the board level¹ more efficiently.

BILBO combines the basic features of scan design with those of signature analysis.² The shift registers form feedback paths by XOR-ing some outputs from the flip-flops and connecting back to some of the flip-flop inputs. A given width of BILBO that implements a primitive polynomial has the corresponding XOR patterns. One combination of the control signals configures BILBO into a multiple-input shift register for compacting circuit responses when the shift registers contain extra control ports. Today's computer designs contain internal VLSI bus paths with widths of 16 or even 32 bits, while the BILBO design retains a bandwidth of 8 bits. Therefore, we need to redesign the BILBO to accommodate

the new wider bus paths. Bhavsar proposed a family of concatenating polydividers with primitive characteristic polynomials to resolve the unextendable BILBO problem for packaged chips.³

To minimize hardware overhead and design time while maintaining certain state and fault coverages, we recommend a bitwise cascadable built-in tester macro cell with an optimum primitive characteristic polynomial. This keeps the CBIT cell in a design library, allowing circuit and system designers to easily construct the necessary feedback path for their BIST circuitry. Previous work on circular self-testing paths (CSTP)² accomplished cascadability, however, by simply connecting registers in a circuit to form a closed loop with a feedback polynomial of x^N+1 . With a nonprimitive feedback characteristic polynomial, the CSTP approach is a special application of the CBIT. The performance of CSTP suffers as a result of its nonprimitive polynomial and requires a sufficiently long testing time when the aliasing probability approximates the asymptotic value 2^{-N} , where N is the input width of the circuit under test.⁴

To further improve testing time, we propose a novel approach based on CBITs that allows concurrent MCM testing. We refer to this strategy as the loop testing architecture. LTA uses CBITs in a pipe interwoven with high I/O-count chips on MCMs. Simulation results show that this guarantees high test coverage with the use of maximum-length pseudorandom sequences for test pattern generation. The aliasing probability compares favorably to that provided by a twofold multi-input linear-feedback shift register⁵ with only a fraction of the area necessary.

MCMs require significant exhaustive testing. The original test vectors for chips with high I/O counts from different manufacturers may not be available for the functional testing of the assembled module.⁶ In this case, parallel-pipelined

exhaustive testing using LTA becomes imperative for the MCM designers to achieve better fault coverage more efficiently than boundary scan. Arrays of CBITs provided to the MCM, either in the form of a small chip on the same substrate or off the MCM test circuitry, allow chips without BIST circuitry to use LTA. Chips with existing on-chip BIST structures easily support LTA.

Cascadable built-in testing structure

The goal of our CBIT design is to provide a macro cell in the design library that expedites the BIT design process. We cascade CBIT cells to form a CBIT suite, using multiplexers and XORs placed in strategic locations to construct different feedback paths. This generates primitive polynomials in a multiple-byte configuration. A CBIT suite, with feedback connections that represent a primitive polynomial, acts as a maximum-length PRS generator.⁷ Not only does CBIT perform test pattern generation and signature analysis, it also permits cascadability to generate a maximal length PRS. In performing signature analysis, a primitive characteristic polynomial gives a quicker convergence of the smaller asymptotic-aliasing probability for a given test length.⁸

The CBIT design. A modified 8-bit BILBO forms a CBIT cell. It consists of three control signals⁹ (C_x , C_y , and C_z), eight parallel inputs (D-bus), eight parallel outputs (Q-bus), an LFSR consisting of eight flip-flops, and XORs that provide the feedback path of the LFSR. Two serial data ports, Scan_in and Scan_out, form the scan path. Finally, Feedback_in and Feedback_out provide the cascading links among CBITs. Figure 1a (next page) shows the 8-bit CBIT cell, while Figure 1b represents a 16-bit CBIT suite configured from two CBIT cells.⁹

Use of the feedback pattern and the generating polynomial guarantees maximum-length PRS generation in both the 8- and 16-bit cases. Notice that in the 16-

Frequently used acronyms

BILBO	Built-in logic-block observer
BIST	Built-in self test
BIT	Built-in test
CBIT	Cascadable built-in tester
CSTP	Circular self-testing path
CUT	Circuit under test
GLFSR	Generalized linear-feedback shift register
LFSR	Linear-feedback shift register
LTA	Loop testing architecture
MISR	Multiple-input shift register
PRS	Pseudorandom sequence
PSA	Parallel signature analyzer
SUT	System under test
TPG	Test pattern generator/generation

bit case, the feedback path for the least significant CBIT suite differs from the path of the most significant CBIT suite (Figure 1b). To guarantee the maximum randomness and quick convergence to the asymptotic value of the aliasing probability, the generating polynomial for the 16-bit CBIT must be prime.⁸ In general, cascading CBITs make extended-length MISRs fit the increasing size of the data buses without redesigning the detail of the BIST circuitry. This speeds up the design modification cycle by making the original designs more testable.

Operation modes provided by CBIT. CBITs provide three modes of operation (Figure 2): parallel-register, scan path, and MISR. During normal operation, the parallel-register mode remains active. CBITs form pipelined parallel registers in the data path. During initialization and signature readout, the scan-path mode becomes active. The Scan_in port shifts in nonzero seeds and the Scan_out port reads out signatures. In addition, a scan path can form through the pipe to read out signatures

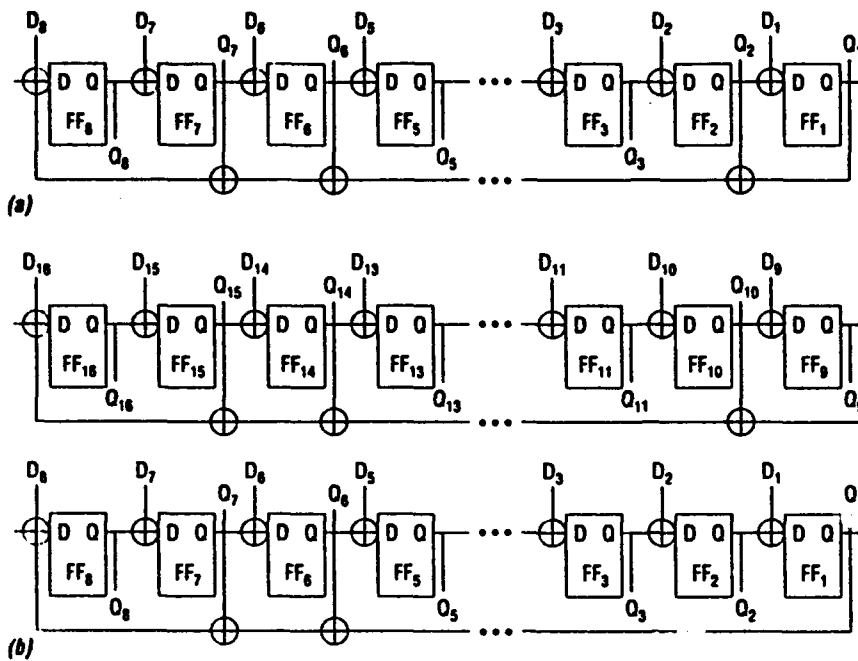


Figure 1. An 8-bit CBIT/MISR with generating polynomial $= x^8 + x^6 + x^5 + 1$ (a); 16-bit cascaded CBIT/MISR with generating polynomial $= x^{16} + x^{14} + x^{13} + x^9 + x^6 + x^5 + 1$ (b).

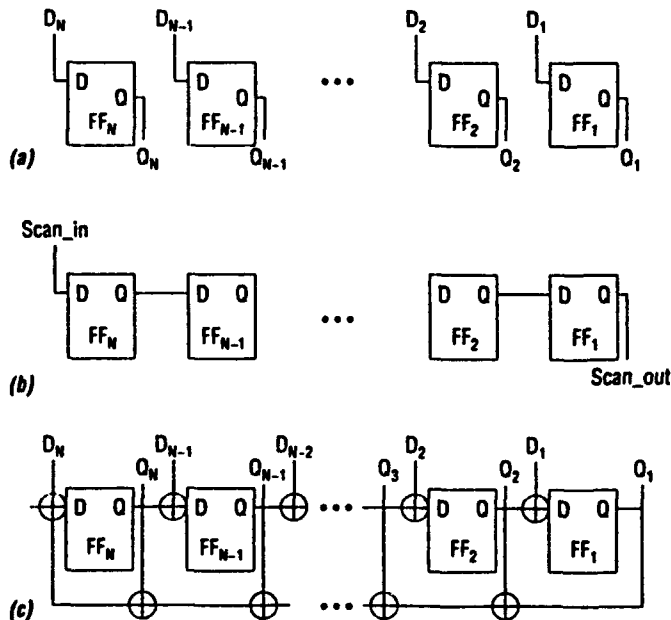


Figure 2. Parallel-in/parallel-out register/buffer mode (a); scan-in/scan-out shift register mode (b); parallel-in/parallel-out LFSR mode (c).

Table 1. Control signals and the corresponding CBIT operation mode settings.

Control signals (C_x, C_y, C_z)	Configuration
(1 1 1)	Parallel register mode
(0 1 -)	Scan path mode
(1 - -)	MISR mode
(1 0 1)	Most significant byte for cascading
(1 1 0)	Least significant byte for cascading
(1 0 0)	Single-byte MISR

in the intermediate stages. Configuration of the CBITs for testing occurs in the MISR mode that concurrently performs pseudoexhaustive test pattern generation for the succeeding CUT and output signature analysis for the previous CUT. The combinations of the control signals C_x , C_y , and C_z provide three major operations as summarized in Table 1. As shown in the last three rows, the combinations of C_y and C_z enable CBIT cascading.

Pipelining for self-testing

The horizontal extension of the CBITs accommodates large I/O MCM testing. Further reduction in testing time results when several functional blocks in an MCM form a pipe to test blocks concurrently. (We refer the functional blocks to those CUTs in a SUT and modules to CUT/SUT with BIT circuits.) Each pipe consists of one zero-stage CBIT suite and subsequent stages of block and CBIT set.

Clusters of functional blocks possessing similar numbers of inputs/outputs form a pipe. Pipes constructed according to their functionality and data width improve efficiency. We then construct CBIT suites to match the data width of each pipe. For CUTs with very limited outputs (for example, encoders), we can cluster a greater number of CUTs for

the CBIT suites to analyze at each stage. Alternatively, the partition/segmentation process discussed by Srinivasan et al.¹⁰ constructs several shorter or narrower pipes. Figure 3a shows construction of pipes for a data path in the SUT, while Figure 3b shows the pipe construction for a control path (which usually has nonuniform I/O bit width or branched signal flows). The requirements on the state and fault coverage and the aliasing probability determine the proper length of any given pipe.

Our previous work⁹ shows preliminary results. Following formation of the pipes, rearrangement of the number of stages in each pipe occurs so that most of the pipes finish self-testing simultaneously. Normally existing data paths with pipelining form naturally self-testing pipes. When the pipe becomes too long and requires decomposition, only the zero-stage CBIT suite adds to the second pipe. Creating all pipes under this guideline after the rearrangement phase gives the maximum parallelism.

For high fan-in CUTs, decomposing the original network into segments with fewer fan-ins¹⁰ reduces computational effort. The controllability, detectability, and observability measures of a segmented circuit remain the same as for the unsegmented CUT.¹¹ Adopting algorithms proposed by Yeh et al.¹² allows grouping of segments into clusters. Often, these clusters identify natural pipes.

LTA. Because the cumulative test results degenerate over multiple stages, we need higher test coverage and lower aliasing probability in the pipelined MISR operation. Further cascading of CBITs in neighboring stages via the Feedback_in/Feedback_out lines increases the length of the CBIT suite. Constructing the scan lines to ease serial scanning of signatures from all CBITs after the test session results in LTA.

Figure 3a also shows the construction of LTA with Feedback_in selected for the CBIT that performs the most significant unit analysis and Feedback_out selected

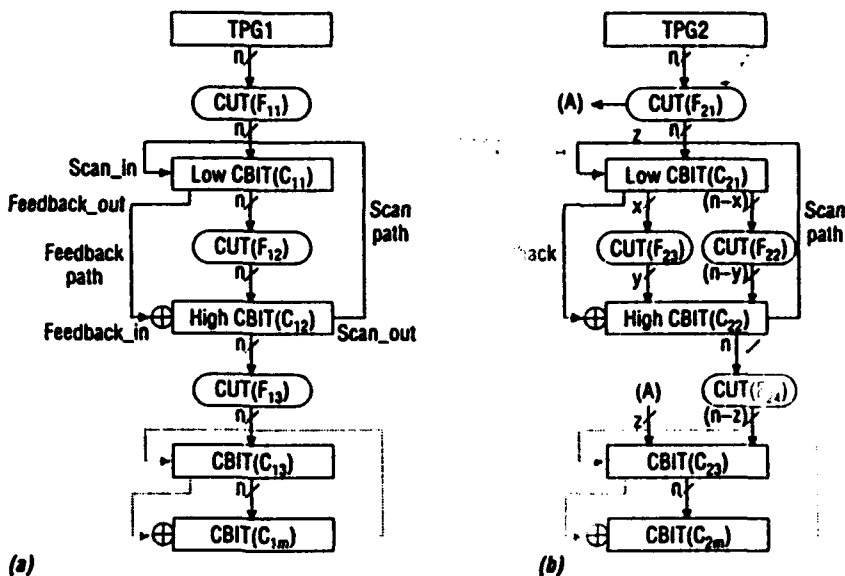


Figure 3. LTA pipe for cuts with: homogeneous data width (primarily data paths) (a); heterogeneous data width (primarily control paths) (b).

for the least significant CBIT. Daisy-chaining the Scan_in and Scan_out ports of the CBITs at each stage gives a scan path for the initialization and scans out the final signature of each CBIT. The last single CBIT suite of a pipe connects to the zero-stage CBIT. Thus for each pipe, we have double-length CBIT suites for signature analysis resulting in smaller aliasing probability for the whole pipe. Figure 4 shows the equivalent data flow when the CBITs are paired to do a double-length signature analysis. The grayed functional blocks (F1, F2, and F4) show the paired testing flow when we cascade two CBITs in LTA. Neighboring CBITs of arbitrary length created with LTA result in a desirable aliasing probability.

Evaluation of the pipelining test

We configure all of the CBITs as MISRs for testing. They generate test patterns and perform signature analysis. Two assumptions justify this decision:⁷

1. The result of the input (seed) and the current state under the operation governed by the characteristic

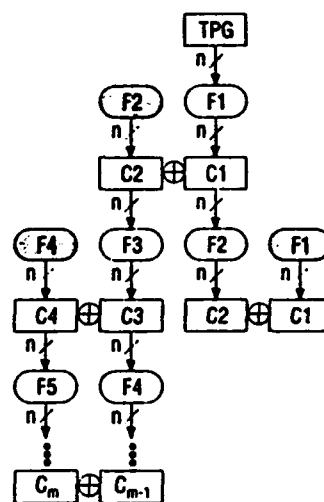


Figure 4. Data path of the m-stage pipelined extended CBIT testing.

polynomial of the LFSR/MISR shall not be 0 for any state of the PRS.

2. Multiple inputs (seeds) to the LFSR/MISR still traverse all the states of the PRS; we do not consider the degeneration or missing of some states in

the PRS because of special combinations or sequences of the seeds.

These assumptions are validated by Kim et al.¹³ who discuss the existence of properties pertaining to the randomness of the patterns generated by a MISR that exist even if the inputs are not equally probable.

To justify the pipelined LTA approach, we must prove two things regarding the dual use of the intermediate CBITs. First, we must show that these CBITs perform effectively as TPGs with patterns generating maximum-length PRS at each stage. The pseudorandom property of the generating polynomial of the CBIT in the MISR mode supports this. (The percentage of the corresponding maximum-length PRS indicates the performance level.) Second, we must show that the limited output patterns of these functional blocks do not disturb the randomness of the signature, where the aliasing probability remains acceptably small after a number of stages.

Properties of the pseudorandom test pattern generation from CBIT. The construction of CBITs by LFSRs with primitive and irreducible characteristic polynomials gives them the following major properties:⁷

- Every element (or state) α in the PRS generated by the LFSR has a complementary element (or state) $\bar{\alpha}$ in the same sequence such that $\alpha + \bar{\alpha} = 0$ (N -bit-wide 0's), where "+" represents the operation on the complementary elements of the PRS as defined by the characteristic polynomial of the LFSR.
- For the cyclic PRS, more than one input seed will either decompose the original maximum-length cycle to several subcycles or merge at least two subcycles together.
- The total number of (distinct) states of all the subcycles (if decomposed by multiple seeds) are $2^N - 1$ for an

N -stage LFSR. In this manner, we exclude the 0 state from the PRS and it forms a trivial cycle ($0 \rightarrow 0$) for the LFSR.

Cascaded CBITs generate test patterns for different functional blocks in a pipe, giving rise to the following observations:

- Each CUT with N -bit-wide input buses needs $2^N - 1$ different test patterns to finish the exhaustive self-testing.
- To exhaustively test the paired CUTs (each with no more than N inputs), one pair of CUTs should generate $2^{2N} - 1$ test patterns.

For example, given an 8-input CUT, we need $2^8 - 1$ test patterns. Assuming that there is no correlation between the two neighboring CUTs for one pair of cascaded CBIT suites, we need one maximum-length cycle of the 8-bit-wide PRS to fully test one 8-bit CUT. However, because of the equally distributed ones in the 16-bit-wide PRS,⁷ we should fully test two 8-bit CUTs before the extended PRS reaches its maximum-length period (which is $2^{16} - 1$). The actual number of the test patterns needed to fully test m CUTs simultaneously using m cascaded CBITs depends on the characteristic polynomial of the extended CBITs and the input seeds. But in general, we have the following relation:

$$(2^N - 1) \leq L \leq (2^{mN} - 1) \quad (1)$$

where L represents the test length needed to test m CUTs exhaustively given m cascaded CBIT suites. Therefore, CBITs perform effectively as TPGs when we choose an appropriate test length according to Equation 1.

Aliasing probability for single-stage MISRs. A special case of the generalized LFSR⁵ occurs with CBITs in the MISR mode. A generalized m -stage, N -input signature analyzer with linear

feedback patterns built over the Galois Field $GF(2^N)$, GLFSR ($m = 1, N$) becomes known as an N -input MISR when $m = 1$. Therefore, when the CBITs have characteristic polynomials designed to be prime, maximum-length test patterns result, and the asymptotic-aliasing probability converges quickly.

Theorem 2 by Karpovsky et al.¹⁴ provides a general formula for calculating the aliasing probability for a one-stage, N -bit-wide MISR:

$$P_{al}(p_0, p_1, p_2, p_3, \dots, p_{2^N-1}) = 2^{-N} \left(\bar{p}_0^L + \sum_{i=1}^{2^N-1} \prod_{j=0}^{L-1} \bar{p}_{i+j} \right) - \bar{p}_0^L \quad (2)$$

where L is the test length and $\bar{p}_0, \bar{p}_1, \bar{p}_2, \dots, \bar{p}_{2^N-1}$ are the Walsh transforms of the error probabilities from an N -bit-output CUT. Error detection does not occur if $p_i = 0$, but always results when $p_i = 1$.

Some closed forms of P_{al} exist with additional conditions.⁵ Two of these closed-form P_{al} 's result from setting the bit-error transition probability p to 0.5:

- When the test length L is $m(2^N - 1)$ (where $m \geq 1$ is an integer), the aliasing probability P_{al} is given by

$$P_{al}\left(\frac{1}{2}\right) = 2^{-N} - 2^{(-mN)(2^N-1)} \quad (3)$$

where $m \geq 1$ is an integer for the independent error model.

- When the number of test patterns is an arbitrary positive integer L and the probability of an output bit being wrong is 0.5, P_{al} is

$$P_{al}\left(\frac{1}{2}\right) = 2^{-N} \left(\frac{1 - 2^{-NL} + (2^N - 1)}{\left(1 - \frac{2^{N-1}}{2^N - 1}\right)^L} \right) \quad (4)$$

for the $2N$ -ary symmetric-channel error model. Notice that for both cases when $2^N \gg 1$, P_{al} converges to an asymptotic

value 2^{-N} . [When the test length L is less than one maximum length for the N -bit-wide MISR for the independent error model (for example, 2^N-1), we use Equation 2 to calculate the exact aliasing probability of the MISR.]

Aliasing probability in the pipelining scheme. In the multistage pipelining MCM testing scheme, we calculate the aliasing probability for the k th stage as

$$\begin{aligned} P_k(\text{aliasing probability at } k\text{th stage}) &= 1 - \left(\text{nonaliasing probability over } k \text{ stages} \right) \\ &= 1 - (1 - P_{al})^k \\ &= 1 - \left(1 - k \times P_{al} + \frac{k \times (k-1)}{2} \times P_{al}^2 - \dots \right) \\ &= \sum_{i=1}^k (-1)^{i-1} \left(\frac{k!}{i!(k-i)!} \right) P_{al}^i \end{aligned}$$

Let $2^N \gg 1$ (generally true for all CBIT suites). We can then approximate P_{al} of the N -input MISRs as 2^{-N} for all the values of m or L in Equations 2 and 3. P_k then simplifies to

$$P_k = \sum_{i=1}^k (-1)^{i-1} \left(\frac{k!}{i!(k-i)!} \right) 2^{-N}$$

By ignoring the contribution from the terms smaller than 2^{-N} , the aliasing probability for the k th stage pipelined CBIT converges to

$$P_k = k \times 2^{-N} \quad (5)$$

Equation 5 gives the asymptotic value for both the symmetric-channel error model with any test length and the independent error model with a test length of at least one maximum length. When the number of stages k is much smaller than the maximum length of the PRS generated by the MISR (for example, 2^N-1), the aliasing probability at the k th stage MISR in the pipelining scheme is of

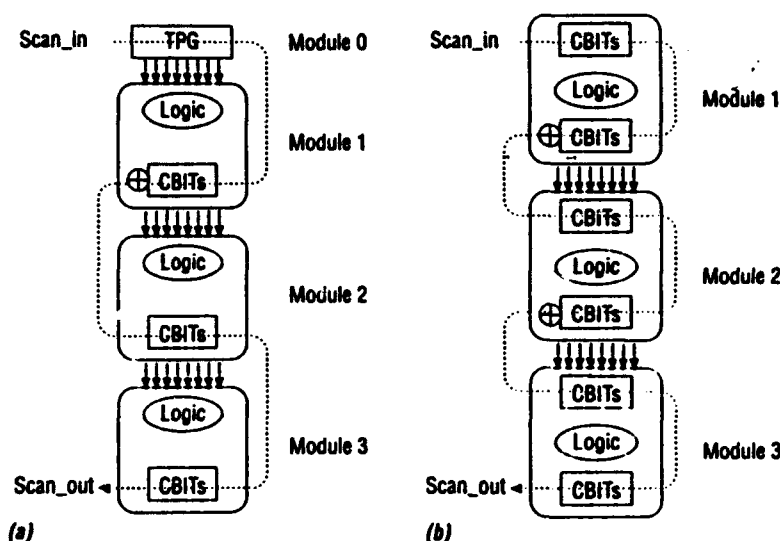


Figure 5. LTA capability for testing: module functionality (a); module functionality and interconnections (b).

the order of magnitude $O(2^{-N})$. The simulation result in Lin and Kaseff,⁹ where the aliasing frequency and probability both stay constant as $O(2^{-16})$ over six stages in the pipelining path, also validates this result.

Thus, the randomness of the signature is preserved in the case of a limited number of multiple inputs. Also, the aliasing probability is sufficiently small given that the number of stages k is small compared with the maximum length of the PRS.

Other LTA applications

LTA has several additional applications. It can test interconnections, and is effective in locating faults in intermediate CBITs.

Capability for testing interconnections. We view interconnections among the MCMs as simplified CUTs with compatible data paths, and the whole system as many CUTs (including the interconnections) requiring testing under the LTA scheme. The pipelining scheme tests interconnections among MCMs by integrating two sets of CBITs next to the I/O pins in each module. The first CBIT set oper-

ates in the MISR mode, validates the input before the signals reach the internal logic, and serves as the TPG for the internal logic blocks. The second set of CBITs also operates in the MISR mode, examining the output from the internal logic circuitry, and serves as the TPG for the interconnection to the next module.

Figure 5a shows one CBIT suite placed at the primary outputs of each CUT with the zero stage CBIT suite added to generate the pseudorandom test pattern for the first CUT. Cascading neighboring CBITs (as shown in Figure 3a) tests the modular functionality of each CUT. However, this implementation cannot test the interconnections between the CUTs. Figure 5b shows an extra CBIT suite inserted near the primary inputs of each CUT. Therefore, we always have a CBIT suite testing either a functional block or an interconnection pattern between two CUTs.

This implementation provides a general approach that can test fault patterns in all permutations, including multiple stuck-at faults, bridging or coupling, and pattern-sensitive faults. The N -bit-wide interconnection network often realizes fewer than 2^N-1 different patterns for

implementing signal links between any two CUTs. However, our N -bit-wide CBIT suite can generate $2^N - 1$ different test patterns to exercise the N -bit-wide interconnection exhaustively.

In scheduling the testing for the interconnections, extra modes become unnecessary. In addition, timing conflicts do not exist because we use two sets of CBITs near the I/O pins that transform the interconnections into another type of CUT directly. This pipelining scheme allows for concurrent testing of both the modules and interconnections. Adapting LTA for interconnection testing by inserting one extra CBIT suite near the input ports (making the interconnections observable) results in area overhead. In contrast, pipeline testing for the modules requires only one CBIT suite at the outputs of each module. Testing the interconnections and the module functionality concurrently requires an extra CBIT suite but saves separate modes for reconfiguring the SUT to test the interconnections. Therefore, simultaneous testing yields significant time savings with only a small area penalty. Furthermore, the placement of CBIT sets holds when we move the I/O ports to the center of the modules.

Fault location in intermediate CBITs. CUT Signatures read out when we configure CBITs in the scan-path mode. Generally, a wrong final signature indicates that faults exist in the test pipe. However, there exists the possibility of faults from different CUTs in a pipe canceling each other and producing a good signature at the last stage. Therefore, we need to know the exact test length applied to each CBIT suite. This results in observable signatures at the intermediate stages and facilitates fault location. By locating faults in specific CUTs, we achieve better fault diagnosis.

Examples

We developed two experiments to demonstrate the effectiveness of the pro-

posed LTA. The first involves testing a homogeneous processor environment consisting of SN74LS181 ALUs. The second encompasses a heterogeneous MCM system with several types of components. We transformed both of these systems into test pipes.

Six-stage ALU pipes

Six ALUs (SN74LS181) form a pipe with 16-bit CBIT suites inserted between the ALUs. Each 16-bit CBIT suite acts as a TPG for the 14-bit input ALU. The 8-bit output of the ALU feeds into a CBIT suite configured for signature analysis. In this experiment, we develop two pipes based on LTA: one implements a primitive characteristic polynomial between the looped CBIT pairs, while the other directly connects the feedback lines without changing the feedback pattern of each CBIT suite. We also reconstruct the straight pipe from our previous experiments⁹ to provide a baseline comparison.

Randomness of the TPG. We measure the randomness of the TPG process at each stage of the three pipes for various test lengths. This allows us to evaluate the effectiveness of the CBITs as TPGs while operating in the MISR mode as well as the impact of pipe length on the test pattern generation. For an N -bit-wide CBIT suite, the randomness measure is 100% if 2^N test patterns are generated. Figure 6 shows the randomness measure for each stage of the three different pipes. In all three configurations, the randomness levels off after the first stage, indicating that the length of the pipe does not affect the random TPG process.

Our previous experiments showed that the required test length L for the two N -bit CUTs under LTA testing (using Equation 1) is smaller than $2^{2N} - 1$. The simulation result in Figure 6 validates this observation by showing that we can exhaustively test all ALUs in each cascading stage of the pipe when L is about

four times the maximum length of the N -bit-wide CBIT suite. That is, instead of $2^{32} - 1$ (or even $2^{28} - 1$) test patterns for the two ALUs, we require only 4×2^{16} test patterns to give a 100% randomness for the two ALUs at each cascading stage.

The cascaded CBIT suite that implements LTA outperforms the straight pipe by producing the best random patterns. As we increase the test length, the CBIT suites eventually generate a 16-bit-wide maximum-length PRS. This validates our earlier assumption that multiple inputs to the PRS generators still produce the maximum-length PRS.⁷ Regardless of how the 8-bit-wide outputs are connected to the CBIT suite (at the higher, lower, or even the middle byte), after a sufficiently long test length (four times of the maximum length) 100% randomness of the 16-bit-wide PRS is still possible.

Aliasing probability of signature analysis. We introduce faults by requiring a single stuck-at-0 fault at the first stage ALU in each pipe. After a specified number of test patterns, we compare the ALU signatures with known good signatures. Aliasing occurs when a faulty pipe produces the same signature as a fault-free pipe.

We compare the aliasing probability at the last stage of the three pipes which have a stuck-at-0 fault at the least significant bit of the first stage ALU output. As shown in Figure 7 (page 46), the aliasing probability of a 16-bit CBIT suite approaches $O(2^{-16})$ as the test length increases in all three pipes. The straight pipe tends to receive aliases early for shorter test lengths. In contrast, a CBIT suite implementing LTA does not exhibit aliasing effects until a sufficiently long test time has elapsed. In this case, aliasing occurs at the sixth stage after 100 tests in the primitive LTA pipe. With test lengths smaller than the length of one maximum-length PRS, the aliasing probability becomes more pronounced. Figure 7b shows that aliasing occurs at the later stages before it shows up in the ear-

lier stages. For the first stage, aliasing occurs after more than 1,000 tests. However, the sixth stage shows aliasing after less than 10 tests. This implies that a warm-up period reduces the aliasing probability at each stage for small test lengths of the pipelined LTA.

In general, CBIT suites implementing LTA exhibit smaller aliasing probabilities than straight pipe CBIT suites. When we use only one CBIT suite in the last stage for comparison, all aliasing probabilities stay at $O(2^{-16})$ (see Figure 7c). If we read the contents of the two CBIT suites as the complete signature, the aliasing probabilities for both pipes implementing LTA become $O(2^{-32})$ in our single stuck fault simulation, a negligible value compared to $O(2^{-16})$. This results from the CBIT suite's extended width of 2^N (2×16 in this case). Note that in Figures 7a and 7c the cascaded CBITs with nonprimitive characteristic polynomials give the same asymptotic value of the aliasing probability as that of the primitive feedback polynomials discussed by Damiani et al.⁸

Area overhead and testing time. The area overhead for implementing LTA consists of the extra wiring required to cascade the CBIT suites with additional XORs (to implement the primitive generating polynomial). As mentioned earlier, constructing the scan path with cascaded CBITs eliminates the need for extra circuitry. The testing time for the LTA pipe is the same as that of the straight pipe. However, with some additional wiring and extended signatures, LTA pipes provide extra observability at each stage in the pipe and a much lower aliasing probability.

Pipes with ALU, caches, and RAM (P-pipe)

Our second experiment involves an MCM (consisting of one SN74LS181 ALU, one 8-bit RAM, and two 16x8 data caches) placed in a four-stage testing pipe. A 16-bit CBIT suite placed at the

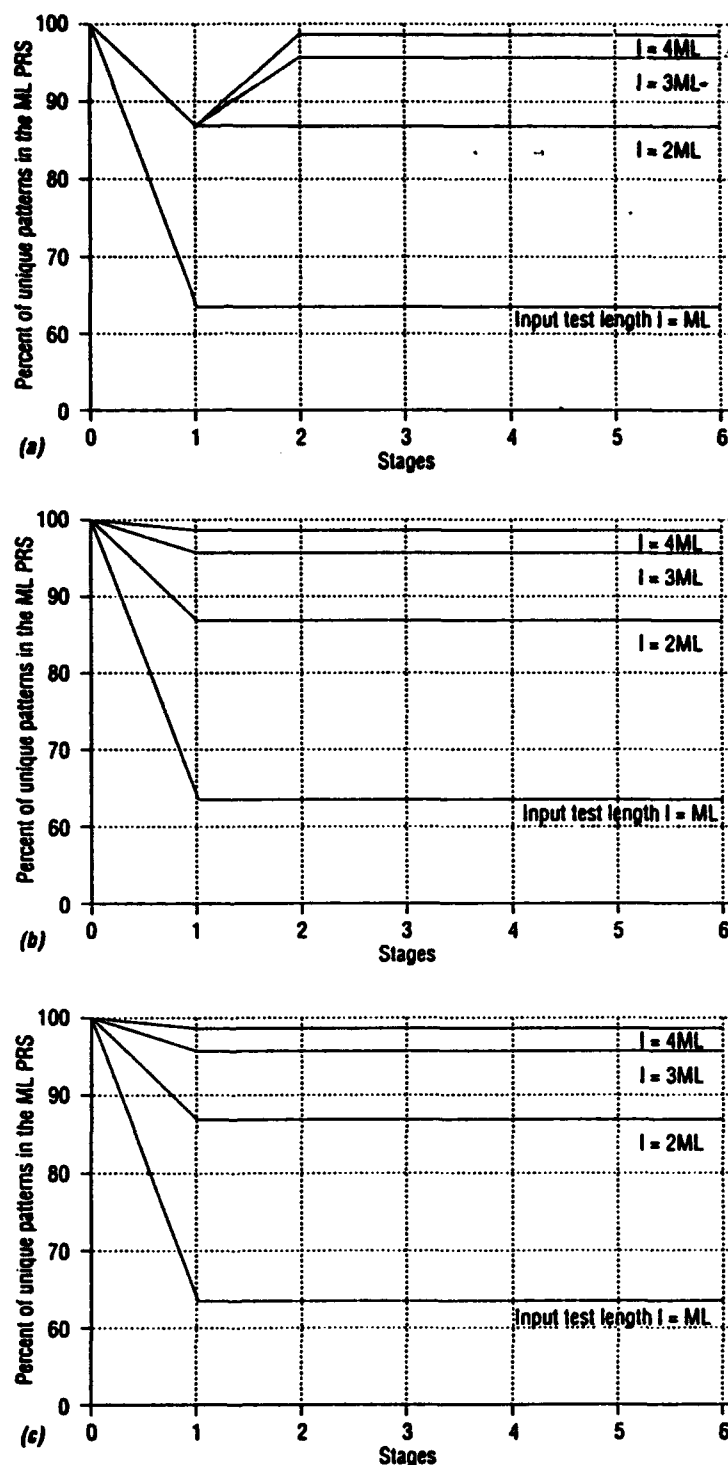


Figure 6. Randomness measure of the CBITs as TPGs over six stages in the ALU pipes ($ML = 2^{16}$): straight pipe (a); cascaded CBITs with nonprimitive polynomial (b); cascaded CBITs with primitive polynomial (c).

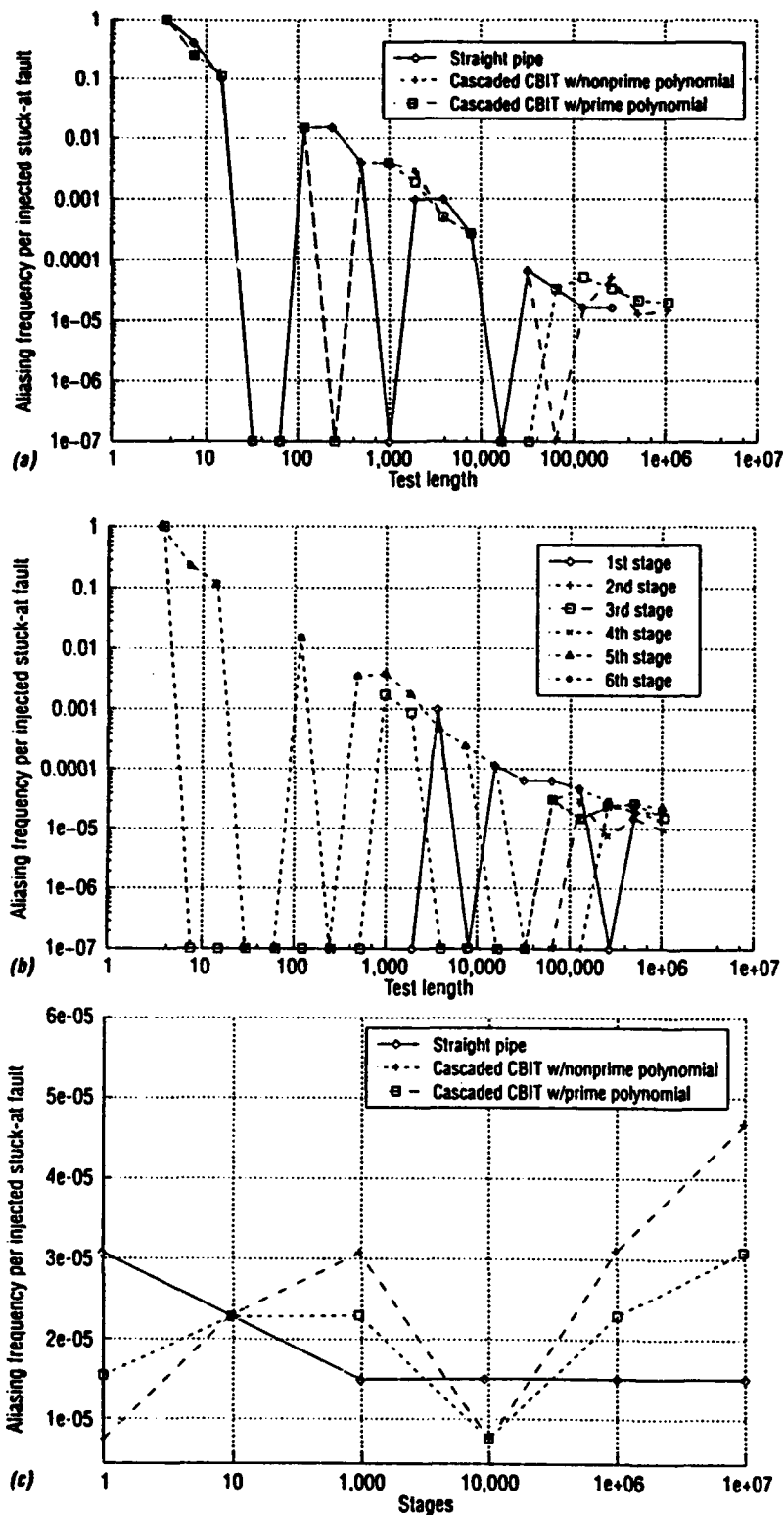


Figure 7. Aliasing frequency in ALU pipes: at the sixth stage for the ALU pipes (a); cascaded CBITs with primitive polynomial (b); test length = 4 ML over six stages (c).

inputs of the ALU acts as the TPG. We insert four 8-bit CBIT cells between the CUTs. Demultiplexing the test patterns from the 8-bit CBIT connected to the inputs of the RAM tests both Address and Data inputs exhaustively. We refer to this as the "straight P-pipe."¹⁵ Adding an extra connection between neighboring CBITs so that paired 8-bit-wide CBITs can perform 16-bit signature analysis for two CUTs simultaneously (Figure 3b) results in the "cascaded P-pipe."

Randomness of the TPG. Figure 8 shows the randomness measure of each stage with different test lengths for the straight P-pipe. In Figure 8a, later stage CBITs reach 100% randomness when the input test length becomes greater than 28 for the 8-bit analysis. In Figure 8b, the zero stage CBIT gives 100% randomness for the 14-bit-wide input bus to the ALU after the input test length becomes greater than four times the maximum length. This finding shows consistency with the previous experiment that warming up the zero stage CBIT improves the quality of the TPG. Figure 9 shows the behavior of the cascaded P-pipe, which is similar to that of the straight P-pipe. This reaffirms the accuracy of our earlier assumption that the maximum-length PRS generated as the test length is at least four times the maximum length for a given data width of the CBIT suites.

In Figures 8a and 9a, LTA requires just four times the maximum-length PRS generated by one CBIT suite to test neighboring CUTs (4×2^8), as opposed to the length it requires from a double-width CBIT suite (2^{16}). The cascaded CBITs show a quicker rate of convergence than a straight pipe. Again, this is similar to the results of the previous ALU pipes.

Aliasing probability of signature analysis. We inject the single stuck-at-0 fault to the least significant bit of the ALU's output. Calculation of aliasing probability occurs by comparing signatures from the faulty pipe with those from a fault-free

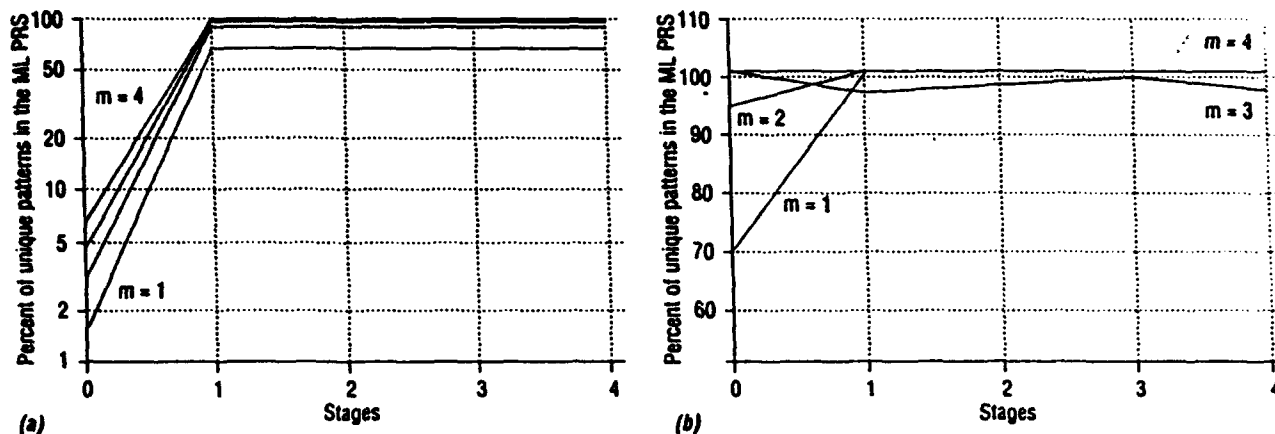


Figure 8. Randomness measure of the CBITs as TPGs over four stages for the straight P-pipe: test lengths are multiples of 256 ($L = m2^8$) (a); test lengths are multiples of 16,384 ($L = m2^{14}$) (b).

pipe. Figure 10a (next page) shows the aliasing frequencies per injected fault of the final signatures in the last stage CBITs of the two P-pipes. Reading the signatures byte-wise from each CBIT cell allows us to calculate the aliasing frequency. We also analyze complete signatures for the extended CBIT pairs. All aliasing frequencies per injected fault of the 8-bit-wide signatures converge to the asymptotic value (2^{-8} for the 8-bit case). However, with a test length smaller than one cycle of the maximum-length PRS (2^8), the cascaded P-pipe gives a smaller byte-wise aliasing frequency than the straight P-pipe. Aliasing does not occur for the extended signatures until the test length reaches one maximum length for the 16-bit signature analysis (2^{16} or 65,536).

Figure 10b shows the aliasing frequencies for the intermediate stages in the P-pipe with cascaded CBITs. The last stage still gives the worst analysis result of the ALU pipes. Once again, warming up the P-pipes seems to improve the test quality. For comparison, we show the aliasing from extended signature analysis ($O(2^{-16}) \sim 1.52 \times 10^{-5}$ for the complete 16-bit-wide CBIT suite).

In these experiments, we analyze the randomness of the TPG and the aliasing problem for multiple-stage parallel sig-

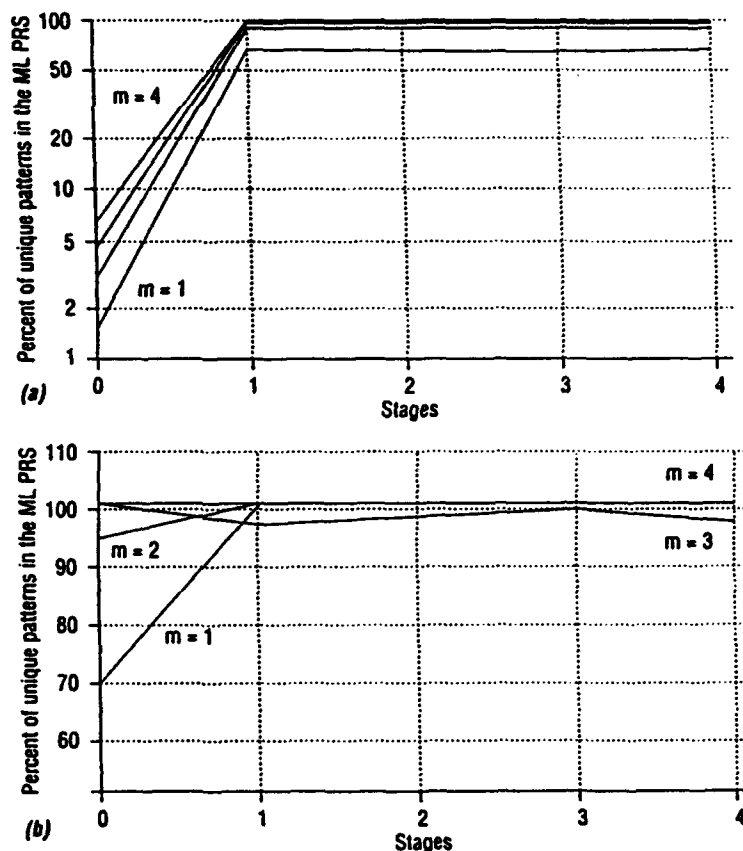


Figure 9. Randomness measure of the CBITs as TPGs over four stages for the cascaded P-pipe: test lengths are multiples of 256 ($L = m2^8$) (a); test lengths are multiples of 16,384 ($L = m2^{14}$) (b).

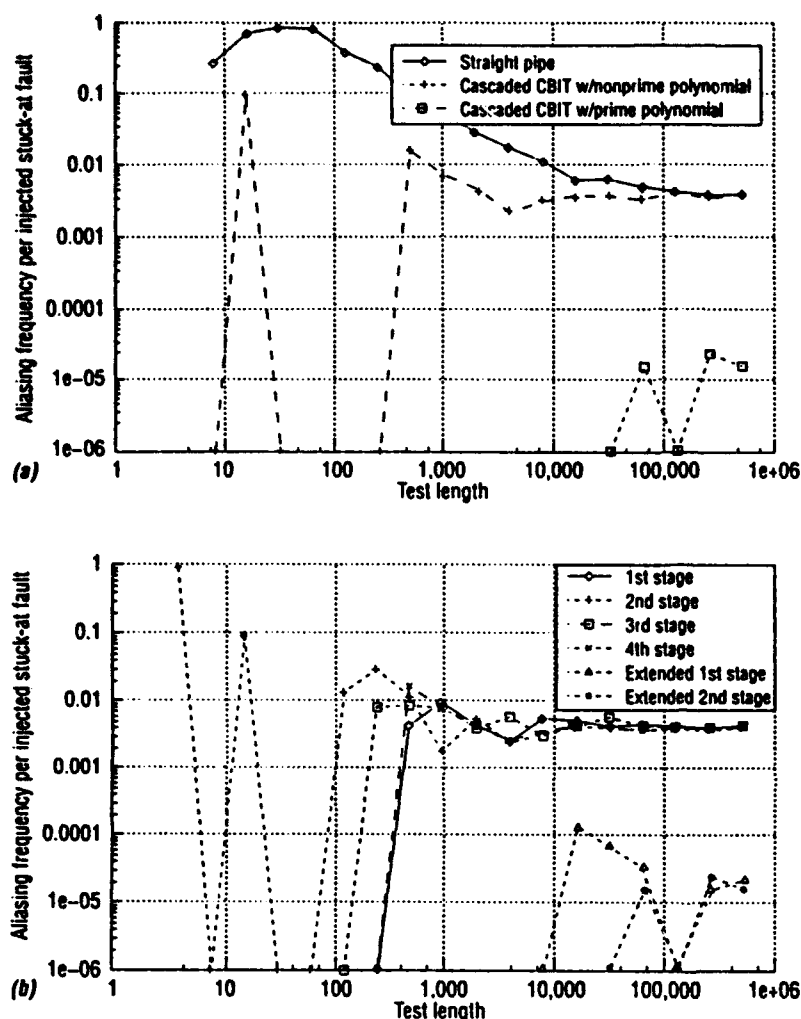


Figure 10. Aliasing frequency: fourth stage for two P-pipes (a); cascaded CBITs with primitive polynomial in the P-pipe (b).

nature analyzers. As the input test length traverses the entire cycle of the maximum-length PRS provided by the MISRs, it visits all states in the PRS at least once. Thus we have 100% randomness of the maximum-length PRS during the TPG process. This also applies to the downstream stages in one pipe. For a well-partitioned pipelined testing path, the aliasing probability is of the same order as that for the serial signature analyzer.

Area overhead and testing time. To implement LTA in the cascaded P-

pipe from the straight-P pipe, the only extra wiring needed is that which connects the CBIT suites. To have a primitive polynomial for the extended CBIT suite, we provide spare XOR gates for the most and least significant suite configurations. Again, we require only two modes for initializing the CBITs and MISR mode analysis.

Performance comparison with other approaches

Two aspects make LTA with CBITs comparable with other testing ap-

proaches: testing time and area overhead. These approaches include the IEEE 1149.1 boundary scan standard (JTAG) and a pipelined BIST with conflict scheduling. We calculate testing time by adding the set-up time (T_{setup}), the module testing time (T_{module}) and the read-out time (T_{readout}). Note that we normalize these times to the average testing time per module. Adding 40 to 80% for wiring to the required hardware components provides an estimate of the area overhead.

LTA does not require different CBIT cells in the design library to test different widths of the data paths. For a wider data bus, we can cascade the CBITs to get an extended PSA without downgrading the quality of the signature analysis. LTA thus eliminates the hardware penalty required by different sizes of BILBOs. (The only hardware overhead needed by LTA is the zero-stage CBIT for a new pipe since the wiring for the cascaded case is negligible.) According to previous performance analyses of CBITs versus the boundary-scan method,¹⁵ CBIT takes less than 10% of the testing time and requires less than twice the area of boundary-scan designs. In both cases, the fault-coverage is 100%. In our examples, we saw that in the six stages of the 16-bit pipelined CBIT suites, the aliasing frequency and probability stay as low as $O(2^{-16})$ for a sufficiently long test length. Therefore, with a limited area penalty and an order of magnitude improvement of the total testing time, LTA drastically reduces the cost of MCM testing in today's competitive market.

Comparing LTA to boundary scan. The boundary-scan approach needs two separate modes and carefully selected test patterns to test the processor for the interconnection failure.¹ When the bit width of the communicating data path increases, the boundary scan requires more complicated test patterns and test cycles.

The original test vectors may not be

available for boundary-scan testing in MCMs using automatic test pattern generation.⁶ Therefore, an N -input CUT requires $O(C \times 2^N)$ test patterns for pseudorandom testing. $C \geq 1$ is a constant given by a statistical estimation on a specific test pattern generation technique.¹⁶ The optimized value for C is 1. However, to have confidence that the most difficult to detect faults are covered, a larger C is required.¹⁶ The total time needed for one N -input CUT under boundary scan testing is given by

$$T_{\text{setup}} + T_{\text{module}} + T_{\text{readout}} \\ = (C \times 2^N) \times (t_{\text{setup}} + t_{\text{module}} + t_{\text{readout}}) \quad (6)$$

where t_{setup} , t_{module} , and t_{readout} are the Scan_in, one execution, and Scan_out time for one CUT. However LTA gives

$$T_{\text{setup}} + T_{\text{module}} + T_{\text{readout}} \\ = t_{\text{setup}} + (4 \times 2^N) \times t_{\text{module}}/k + t_{\text{readout}} \quad (7)$$

where 4×2^N is the maximum given by Equation 1 when $m = 2$ and k is the total number of stages of a pipe implementing LTA. Thus, LTA saves time for scan-in, scan-out, and pseudoexhaustive testing per module/CUT, especially when $4 \ll k \ll 2^N$. For example, if $t_{\text{module}} = t_0$ clock cycles, $t_{\text{setup}} = 16$ clock cycles and $t_{\text{readout}} = 16$ clock cycles for a 16-bit input/16-bit output CUT, then the boundary scan will need $65,536C(32 + t_0)$ clock cycles to finish the pseudoexhaustive testing. LTA requires $32 + 32,768 t_0$ clock cycles for eight stages.

LTA does not require extra testing time in a separate mode for interconnection testing. (In contrast, boundary scan does.) So, the total testing time per module and its interconnection remains

$$O(t_{\text{setup}} + 2^N[t_{\text{module}} + (t_{\text{interconnect}} - 0) + t_{\text{readout}}])$$

since LTA can test both the interconnection and the processor logic at the same time. Here, $t_{\text{interconnect}}$ represents the time for signals to transfer through the inter-

connection network. Its value is negligible compared to the other t 's in the formula. However, if the interconnections have a long propagation delay, we cannot ignore $t_{\text{interconnect}}$. For boundary scan, the total testing time per CUT with its interconnections is given by

$$(C \times 2^N) \times [t_{\text{setup-all}} + t_{\text{module}} + (t_{\text{interconnect}} - 0) + t_{\text{readout-all}}]$$

where $t_{\text{setup-all}}$ and $t_{\text{readout-all}}$ represent the sums of scan-in time and scan-out time, respectively. The time savings over boundary scan again indicates the efficiency of LTA during interconnection testing.

Considering the area overhead, both LTA and boundary scan need five extra electrical pads for one processing element. However, the control logic of boundary scan requires four I/O pins (test-mode-select, test-reset, scan-in, and scan-out).

Due to the XOR gates required to implement the extended generating polynomial, LTA consumes more area than boundary scan does. Our previous experiment¹⁵ in testing the MCM shows that LTA takes 4,240 transistors in the four-stage straight P-pipe case and boundary scan takes 3,040 transistors in total. Therefore, in this example LTA consumes about 39% more area than the boundary scan. However, with limited area overhead LTA provides a superior BIT implementation with improved state coverage and exponentially lower aliasing probability. Furthermore, LTA provides the extensibility for the PSAs in terms of the CBIT implementation and also the pipelining for several CUTs to be tested concurrently.

Comparing LTA to pipelined BIST with conflict scheduling. Other pipelined BIST approaches in Abadir and Breuer¹⁷ alternate modes of TPG and PSA in one LFSR circuit. Krasniewski and Albicki's implementation¹⁸ gives one-stage analysis for all the CUTs in a

pipelined data path by centralizing the TPG and distributing the PSA (for example, using one set of BILBOs as TPGs for all pipes with a one-stage BILBO-CUT's BILBO structure and separating outputs to several PSAs). Establishing conflict tables makes sure that the LFSRs perform TPG and PSA separately in different testing schedules. We call this approach pipelined BIST with conflict scheduling. The total averaged testing time for one N -input CUT ("kernel"¹⁷) is given by

$$T_{\text{setup}} + T_{\text{module}} + T_{\text{readout}} \\ = t_{\text{setup}} + t_{\text{module}} + (2^N - 1) \times D + t_{\text{readout}} \quad (8)$$

Here 2^N represents the number of test patterns¹⁷ and D represents the latency between the current TPG and its immediate predecessor. The minimum (or optimized value) for D is one clock cycle. Usually D is greater than one clock cycle because generation of the new test pattern cannot occur until the system loads the previous pattern to the CUT/kernel as soon as the bus is available.

By comparing Equations 7 and 8, we see that the conflict table approach requires more testing time than LTA, even though we set the optimized pipelining schedule for the best value of D . This occurs because our LTA approach uses the fundamental characteristics of the MISRs to operate simultaneously as TPG and PSA. Thus LTA eliminates the waiting time for the available register and bus to generate a separate test pattern for the CUT/kernel. In addition, the possibility that additional MISR circuits or interconnections are required by the conflict scheduling to separate TPG and PSA modes does not occur in LTA. A 16-input/16-output CUT needs a testing time of at least $65,567 + t_0$ clock cycles (calculated from Equation 8 with $t_0 = t_{\text{module}}$) which is greater than the $32 + 32,768 t_0$ clock cycles given by the 8-stage LTA (t_0 is at most one clock cycle).

Due to the dual TPG/PSA mode provided by LTA, we reduce the exhaustive testing time by cutting the time required for scheduling conflicts on one MISR. In addition, the horizontal (for bit-size changes) and vertical (for multiple CUTs to be tested in a pipe) extensibility given by LTA provides the best utilization of the parallel testing. Therefore, we can perform pipelining and achieve parallelism on one system with minimal design modification and optimal test scheduling.


Our LTA scheme requires less testing time compared to boundary scan and pipelining with conflict scheduling without losing the effectiveness of the test coverage. We anticipate no significant area overhead (except for the spare XOR paths for cascadability) in LTA. Furthermore, we believe that by rearranging the placement of the CBIT circuits and including test scheduling, we can gain high testability and observability of the permanent faults in both the processor and interconnection.

WE PROPOSED A CBIT to test MCM modules configured in a pipelined fashion. Cascading the CBITs produces high test coverage with 100% randomness in the TPG process and low aliasing probability in signature analysis. In addition, the CBIT circuit can serve as a switching device for module reconfiguration.

We also introduced LTA as a way to reduce aliasing probability. When compared to the GLFSR approach,⁵ LTA gives a similar aliasing probability as the twofold GLFSR. LTA implementation also works when the I/O ports are moved to the center of the chip area in the future system design.

LTA exhibits greater efficiency when we partition the MCM testing sessions into several subcircuits for parallel testing. Yeh et al. discuss partitioning algorithms using netlist as inputs,¹² while Srinivasan et al.¹⁰ propose partitioning at a higher level. Seth and Agrawal discuss

in depth analysis of testability of partitioned CUTs.¹¹

Our future work will emphasize integrating partitioning and clustering algorithms with LTA, to allow automation of hierarchical functional test methodology for MCMs. We expect that modifying LTA and CBIT circuitry to accommodate the interconnection reconfiguration and self-purging in the MCM will improve fault tolerance. 

Acknowledgment

We thank Harry Chen of Cadence Design Systems Inc. for his valuable comments in clarifying some ideas in this work.

References

1. C. M. Maunder and R. E. Tulloss, "Testability on TAP," *IEEE Spectrum*, Feb. 1992, pp. 34-37.
2. B. J. Koenemann, J. Mucha, and G. Zwiehoff, "Built-in Logic Block Techniques," *Proc. Int'l Test Conf.*, IEEE Computer Society Press, Los Alamitos, Calif., 1979, pp. 37-41.
3. D. K. Bhavsar, "Concatenable Polydividers: Bit-Sliced LFSR Chips for Board Self-Test," *Proc. Int'l Test Conf.*, IEEE CS Press, 1985, pp. 88-93.
4. S. Pilarski, A. Krasniewski, and T. Kamada, "Estimating Testing Effectiveness of the Circular Self-Test Path Technique," *IEEE Trans. CAD*, Vol. 11, No. 10, Oct. 1992, pp. 1301-1316.
5. D. K. Pradhan and S. K. Gupta, "A New Framework for Designing and Analyzing BIST Techniques and Zero Aliasing Compression," *IEEE Trans. Computers*, 1991, Vol. 40, No. 6, pp. 743-763.
6. C. A. Pina, "Implementation of an MCM Brokerage Service," *Proc. IEEE MCM Conf.*, IEEE CS Press, 1993, pp. 46-51.
7. S. Golomb, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, Calif., 1982.
8. M. Damiani et al., "Aliasing in Signature Analysis Testing with Multiple Input Shift Registers," *IEEE Trans. CAD*, Vol. 9, No. 12, Dec. 1990, pp. 1344-1353.
9. T. T. Lin and C. Kaseff, "Performance Evaluation of Cascadable Built-In Tester for Large I/O Multichip Modules," *Proc. of the Fourth Ann. IEEE Int'l ASIC Conf.*, IEEE CS Press, 1991, pp. 93.1-93.4.
10. R. Srinivasan, S. K. Gupta, and M. A. Breuer, "An Efficient Partitioning Strategy for PseudoExhaustive Testing," *Proc. 30th IEEE Design Automation Conf.*, IEEE CS Press, 1993, pp. 525-530.
11. S. C. Seth and V. D. Agrawal, "A New Model for Computation of Probabilistic Testability in Combinational Circuits," *Integration, The VLSI Journal*, North-Holland Publishing Co., Amsterdam, Vol. 7, No. 1, April 1989, pp. 49-75.
12. C. W. Yeh, C. K. Cheng, and T. T. Lin, "A Probabilistic Multicommodity-Flow Solution to Circuit Clustering Problems," *Proc. IEEE/ACM Int'l Conf. CAD*, IEEE CS Press, 1992, pp. 428-431.
13. K. Kim, D. S. Ha, and J. G. Tront, "On Using Signature Registers as Pseudorandom Pattern Generators in Built-In Self-Testing," *IEEE Trans. CAD*, Vol. 7, No. 8, Aug. 1988, pp. 919-928.
14. M. J. Karpovsky, S. K. Gupta, and D. K. Pradhan, "Aliasing and Diagnosis Probability in MISR and STUMPS Using a General Error Model," *Proc. 1991 Int'l Test Conf.*, IEEE CS Press, pp. 828-839.
15. T. T. Lin, J. Comito, and C. Kaseff, "Evaluation of Test Strategies for Multichip Modules," *Proc. 5th Ann. IEEE Int'l ASIC Conf.*, IEEE CS Press, 1992, pp. 234-237.
16. J. A. Abraham and V. K. Agarwal, "Test Generation for Digital Systems," *Fault-Tolerant Computing—Theory and Techniques*, D. K. Pradhan, ed., Prentice-Hall, Englewood Cliffs, N.J., 1986, pp. 1-94.
17. M. S. Abadir and M. A. Breuer, "A Knowledge-Based System for Designing Testable VLSI Chips," *IEEE Design & Test of Computers*, Vol. 2, No. 4, Aug. 1985, pp. 56-68.
18. A. Krasniewski and A. Albicki, "Automatic Design of Exhaustively Self-Testing Chips with BILBO Modules," *Proc. Int'l Test Conf.*, IEEE CS Press, 1985, pp. 362-371.



Ting-Ting Y. Lin is an assistant professor in electrical and computer engineering at the University of California, San Diego. Her research interests include fault-tolerant computing, design automation, VLSI testing, fault modeling, and system reliability. She received a BS from National Chiao-Tung University, Hsinchu, Taiwan, Republic of China and a PhD in computer engineering from Carnegie Mellon University. Lin is a member of the IEEE, the Computer Society, and Sigma Xi.



Huoy-Yu Liou is working toward her PhD in computer engineering at the University of California, San Diego. Her research interests include fault-tolerant computing in self-test design methodology for VLSI circuits, and fault modeling of free-space optical interconnections. Previously she was a senior member of the technical staff at Cadence Design Systems in San Jose, California. Liou received her BS in physics from National Taiwan University, Taipei, an MS in physics from the University of Pittsburgh and an MSEE from the University of California, San Diego. She is a student member of IEEE and the Computer Society.

Send correspondence to Ting-Ting Y. Lin at The Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093; lin@celece.ucsd.edu.

Call for Articles

Special Issue on Design and Test in Eastern Europe

IEEE Design & Test of Computers will publish a special issue devoted to Eastern Europe in March 1995. The issue will spotlight design and test research and development activities. Even though the problems of designing and testing complex logical systems may not be of a different nature in Eastern Europe, often their interesting solutions are untold via IEEE publications. This special issue will give *D&T* readers an opportunity to explore the current status and future trends of design and test in Eastern Europe.

Suggested article topics include, but are not limited to, the following:

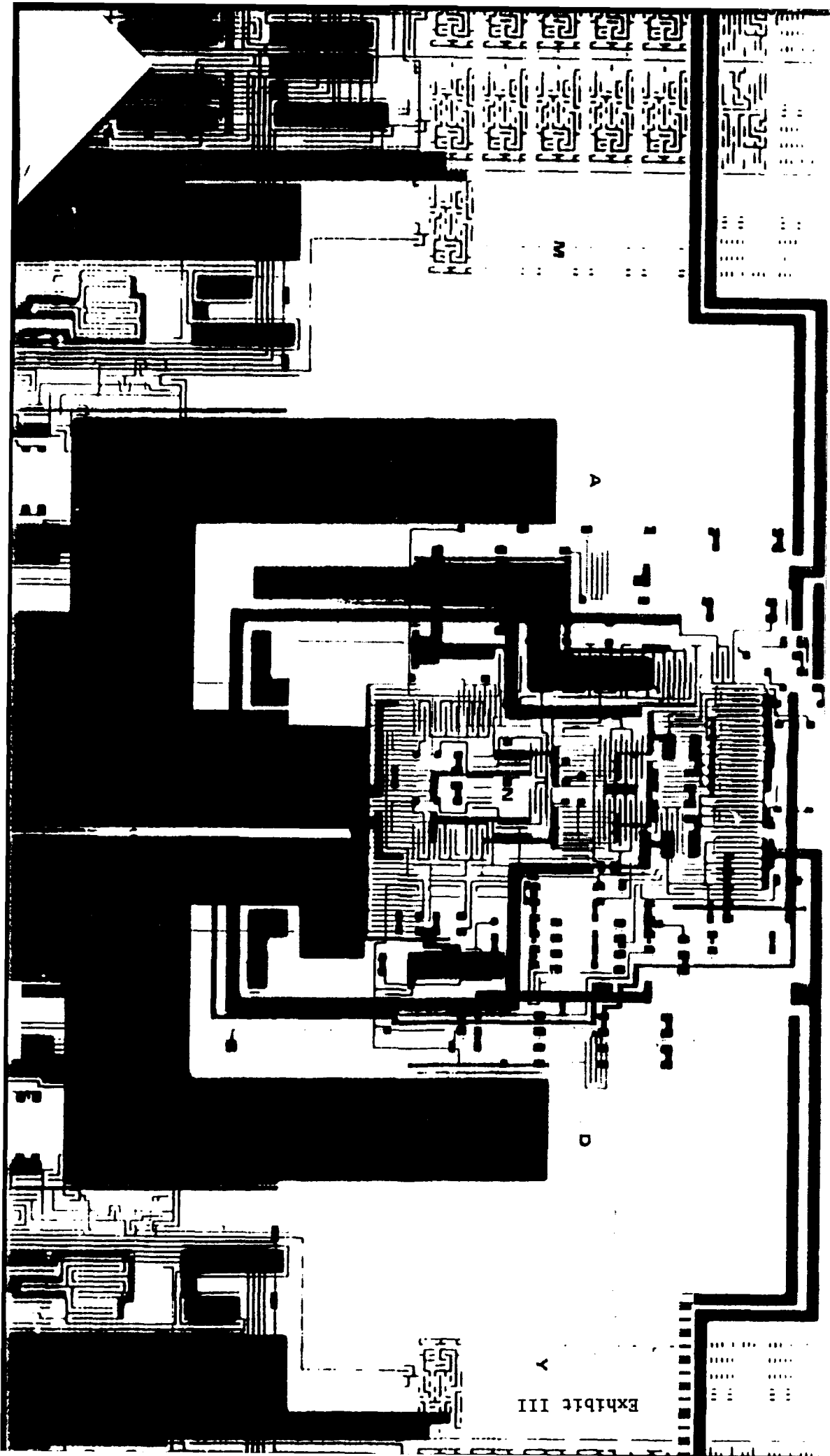
- Analog design & test
- Automatic test generation
- Design for testability
- Fault modeling/simulation
- FPGA architectures
- High-level synthesis
- Memory system design & test
- Multichip module design & test
- Online/offline checking
- Physical design
- Simulation, functional, logic, circuit
- Synthesis for speed, power, area
- Timing analysis and verification
- Wafer-scale integration

Submit by May 1, 1994, six copies of previously unpublished manuscripts of no more than 20 typewritten, double-spaced pages, including all figures and references. Include a title page with article title, full name(s), affiliation(s), physical and electronic mail address(es), telephone number(s), a 300-word abstract, and a list of keywords that identify the central issues in the article. For author guidelines, contact Karen Potes, IEEE Computer Society Publications Office at (714) 821-8380, fax (714) 821-4010.

Submit manuscripts for review to Guest Editors:

Yervant Zorian
AT&T Bell Laboratories
PO Box 900
Princeton, NJ 08540
USA
Phone: (609) 639-3176
Fax: (609) 639-3197
yzy@ohm.att.com

Jan Hlavicka
Czech Technical University
Faculty of Electrical Engineering
Technicka 2, 16627 Praha 6,
Czech Rep. Nic
Phone: +422-311-9903
Fax: +422-24310784
hlavicka@feld.cvut.cz



Circuit Partitioning for Pipelined Pseudo-Exhaustive Testing Using Simulated Annealing

Huoy-Yu Liou, Ting-Ting Y. Lin, Lung-Tien Liu*, and Chung-Kuan Cheng*

Dept. of Electrical and Computer Engineering (0407) and Dept. of Computer Science Engineering*

University of California at San Diego, La Jolla, CA 92093

liou@ece.ucsd.edu, lin@ece.ucsd.edu, and kuan@cs.ucsd.edu

(619) 534-4738 and (619) 534-2486 (fax)

Abstract

A novel approach for partitioning circuits with high fan-ins which are not suitable for pseudo-exhaustive testing is presented. Circuits under test (CUTs) are modeled as directed graphs and cost function is developed for the optimization algorithm. Disjoint circuit partitions are generated not only for reducing the exhaustive test length but also for pipelined testing. Experiments on benchmark circuits demonstrate that simulated annealing produces good results for future applications.

Introduction

Pseudo-exhaustive testing offers 100% fault coverage [16] but suffers long testing time for large fan-in circuits. Partitioning facilitates testing by disconnecting one part of a network from the other, thus making smaller subsystems for built-in self test (BIST). This simple *divide and conquer* strategy has been adopted to reduce the input size of each combinational circuit for improving long testing time.

Partition for complex circuit under test (CUT), using *segmentation cells* was proposed by [1] and [2]. These segmentation cells can be flip-flops added to the circuitry. They can also be generated by moving existing registers in the CUT to the boundary of the partitions and applying re-timing techniques [3] such that circuit timing requirements can be preserved or even improved along with better testability. The controllability, detectability, and observability of partitioned circuits can also be measured at a reduced effort comparing to the unpartitioned circuit [4].

There are several approaches on circuit partitioning for pseudo-exhaustive testing. Output cone size reduction [5] reduces the *dependency* of each output cone of the CUT when the dependency is greater than a user-specified number, r . This may lead to overlapping segments at the expense of potential test schedule conflicts or extra efforts in generating exhaustive test patterns. Udell's [6] gives fewer cut points than a two-way partition and clustering approach [7]. Area overhead, not explicitly represented in the cost function, is introduced by the forward-merging and collapsing the reconvergent fanouts. Bhatt *et al* [2] gives a leveled approach trying to preserve the timing of a CUT by inserting extra segmentation cells. A coordinated circuit partition and testing pattern generation approach by Jone and Papachristou [8] ensured disjoint segments by inserting more registers at the segmental boundary.

In a continuing effort to systemize our pipelined built-in cascable test (CBIT) technique [9] for very large integrated

circuits and systems, we perform partitioning to improve test performance by marking cut points for segmentations. *Simulated Annealing* technique [10] was used to find a global minimal solution under given performance constraints. Disjoint partitioning is proposed for implementing pipelined pseudo-exhaustive testing which has the potential of being fast, efficient, and hierarchical for designing testable circuits. Experiments on the 1985 ISCAS benchmark circuits [11] were performed and the results are encouraging when compared to the approaches in [5][6].

This paper is organized as follows. We first explain our partitioning strategy for pipelined pseudo-exhaustive testing. The theoretical investigation and problem formulation follow to model the CUTs and the cost function for optimization. Simulated annealing algorithm is outlined and its application to find optimal partitioning is briefed. Finally simulation results on ISCAS 85 benchmarks are presented. Discussion on future work follows.

Formulation of the Problem

The partitioning for pipelined pseudo-exhaustive testing can be formulated as the classical m -way partitioning problem with user-specified constraints. To provide a better understanding and easier handling on the disjoint m -way partitioning problem, it is desirable to map the CUT to a graph representation and transforming the constraints into mathematical equations. Thus we can model the problem as the m -way graph partitioning.

Simulated annealing determines the number of partitions, m , after given a big initial value (say 50) by returning empty partitions. Those intermediate segmentation cells can be grouped forming several registers with similar structures as the multiple-input shift registers (MISRs) [12]. In this way, the CUT is setup for a potential pipelined scheduling during the testing mode.

A. Graph Construction

A circuit can be represented as a weighted, directed graph, $G = (V, E)$, where V is the set of vertices/nodes in G and E is the set of edges connecting two nodes in V . Modules or gates in the circuitry are modeled as vertices in the graph, i.e.,

$$V = \bigcup_{i=1}^{|V|} \{v_i\}, \text{ where } |V| \equiv N \text{ is the number of vertices in } G.$$

Signals are modeled as edges with directions from source to sink in the network, e.g., e_{ij} is from node v_i to node v_j . When

the signals are equally weighted, the original graph reduces to a directed graph.

An m -way partition, $\Pi_m: V \rightarrow \{1, 2, 3, \dots, m\}$ and $\pi_i = \{v \in V \mid \Pi_m(v) = i, 1 \leq i \leq m\}$, is an operation such that $V = \pi_1 \cup \pi_2 \cup \pi_3 \cup \dots \cup \pi_m$. Two different partitions, π'_m and π_m are called *neighbors* if there exists a move, O , which is a permutation of the v_i 's in the current partition π_m , $O: \Pi_m \rightarrow \Pi'_m$, such that

$$O(\{\pi_i\}_{i=1}^m) = \{\pi'_i\}_{i=1}^m.$$

Fig. 1 shows an example of the construction of G . The original schematics of a CUT is shown in Fig. 1a. This small test circuit, c17 from ISCAS 85 benchmark, has five primary inputs, six NAND2 gates and two primary outputs. Fig. 1b shows the equivalent graph representation of this circuit.

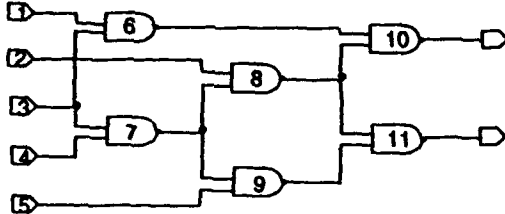


Fig. 1a Schematics representation of a circuit under test (CUT)

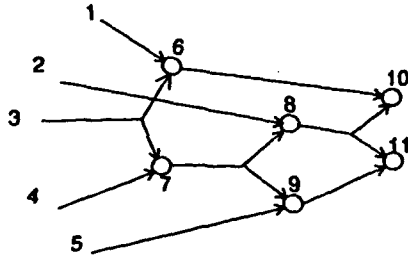


Fig. 1b Graph representation of a circuit under test (CUT)

B. Partitioning with Input Constraints (PIC)

Here we provide a study on the fundamentals for this disjoint m -way graph partitioning problem and prove it to be NP hard. A function, w , was defined as a mapping $w: E \rightarrow I(\text{integer})$ such that

$$w(\pi_i) = \left| \bigcup_{1 \leq j \leq m, (j \neq i)} e_{ji} \right| = \epsilon_i,$$

where ϵ_i is the total number of non-self, incoming edges to partition π_i .

For a given graph, $G = (V, E)$, and an integer, l , we want to partition V into m subsets, $\pi_1, \pi_2, \dots, \pi_m$ with an objective to minimize m under the constraint that $w(\pi_i) \leq l$, $\forall i, 1 \leq i \leq m$.

NP COMPLETENESS: The partitioning with input constraints is NP complete.

Proof:

We first state a 3-Partition Problem which is NP complete. Then we show that a 3-partition problem can be transformed to the problem of partitioning with input constraints.

(i) 3-Partition Problem [17]:

A set V of $3k$ elements, a bound $B \in \mathbb{Z}^+$, and a size

$$s(v) \in \mathbb{Z}^+ \text{ for each } v \in V \text{ such that } B/4 < s(v) < B/2 \text{ and such that } \sum_{v \in V} s(v) = kB.$$

It is NP-complete in the strong sense to partition V into k disjoint sets, i.e., V_1, V_2, \dots, V_m such that for $1 \leq i \leq k$, $\sum_{v \in V_i} s(v) = B$.

(ii) For each member V_i in the 3-Partition Problem, we construct a node v_i with $s(v_i)$ distinct inputs. Suppose $l = B$, if we can achieve $m = k$, we can solve the 3-Partition Problem. Therefore, the PIC problem is NP-complete.

This PIC problem is still NP-hard even when some nodes are duplicated in different partitions. A statement can be drawn from the above proof:

The partitioning with input constraints remains NP-complete even if we allow replicates, i.e., $\pi_i \cap \pi_j \neq \emptyset$.

C. Cost Function

An N input CUT can be tested pseudo-exhaustively when it is divided into m segments according to the constraint that each partition cannot have more than l inputs [16]. The resulting test length for each partition is at most 2^l which is much smaller than the original 2^N . If the partitioned CUTs do not overlap one another, the time dependency for testing those segments is minimal such that the partitions can be tested concurrently. When resource sharing conflicts exist among those m partitions during testing, the worst case is to test partitions sequentially. Therefore, the total testing time, T , for a partitioned CUT is

$$\max \{t_i\}_{i=1}^m \leq T \leq \sum_{1 \leq i \leq m} t_i, \text{ where } t_i \text{ is the total testing}$$

time needed for each segment. T can be as fast as all the partitions tested in one session or be as slow as to test one partition after the other.

The objective of our approach to partition for pipelined pseudo-exhaustive testing can be represented as a cost function on the circuit graph, G . The major concerns in our implementation are the total number of incoming edges to all partitions should be minimal and the total number of CBITs needed (thus the area overhead) is minimized. To avoid the occurrence of big ϵ_i summing with small ϵ_i in the minimized total number of incoming edges, we want to minimize the maximum value of ϵ_i also.

In order to make the penalty of incoming edges to each partition significant and total number of CBITs is minimized, the cost function is defined as

$$C(\{\pi_i\}_{1 \leq i \leq m}) = \sum_{1 \leq i \leq m} \epsilon_i + \sum_{1 \leq i \leq m} e^{(\epsilon_i - l)} + e^{\max \{\epsilon_i\}_{i=1}^m - l} + n \quad (1)$$

, where $n = \sum_{1 \leq i \leq m} \left\lceil \frac{\epsilon_i}{l} \right\rceil$ is the total number of CBITs needed for pipelined pseudo-exhaustive testing.

Simulated Annealing

Simulated annealing has been shown to have good results in solving problems searching for global optimum (or optima) such as constraint partitioning [10][13]. For our m -way partition, we adopt the simulated annealing algorithm [14] as the core algorithm and developed a problem-specific modifications for the whole annealing/optimization process.

A. The Generic Algorithm

Simulated annealing uses a parameter called *temperature* to control the probability of accepting a random *uphill* move. An 'uphill move' is denoted to as the randomly picked solution having a *higher* cost than current solution. This is designed to avoid being trapped in local optima. A *cooling ratio*, γ , where $0 < \gamma < 1$, is to control how the temperature is gradually lowered until the stopping criteria are reached. We let L be the number of random trials for a given temperature, where $L = (\text{size of the neighborhood}, S) \times \text{SIZEFACTOR}$, which gives the size of the problem. There are S different neighboring states for a single move from the current partition. *SIZEFACTOR* is an experimental parameter which gives more chances for the random trials to cover more neighbors. Table 1 shows the generic algorithm of simulated annealing and Step 3 is the *Metropolis loop* [14].

```

STEP 1 Generate a random initial solution  $\{\pi_i\}_{i=1}^m$  and
        calculate its cost,  $C$ .
STEP 2 Choose an initial temperature  $T_{init} > 0$ .
STEP 3 While (stop criteria not met) do
    3.1 Perform the following random trials  $L$  times for one  $T$ :
        3.1.1 Pick a random neighbor of  $\{\pi_i\}_{i=1}^m$ .
        3.1.2 Calculate the cost of the neighbor,  $C'$ .
        3.1.3 Let  $\Delta = C' - C$ .
        3.1.4 If ( $\text{random} < e^{-\Delta/T}$ ), accept  $\{\pi_i\}_{i=1}^m$ .
            (If  $\Delta \leq 0$ , do a downhill move, otherwise accept
             with a probability  $e^{-\Delta/T}$  for the uphill move.)
        3.1.5 Update  $\{\pi_i\}_{i=1}^m$  to  $\{\pi'_i\}_{i=1}^m$  and
             $C$  to  $C'$ .
        3.1.6 If  $C$  is better than  $best\_C$ , update  $best\_C$  to  $C$ 
            and  $\{\pi_i^{best}\}_{i=1}^m$  to  $\{\pi_i\}_{i=1}^m$ .
    3.2 Change  $T$  to  $\gamma T$ .
STEP 4 Return best solution  $\{\pi_i\}_{i=1}^m$ .

```

Table 1: Generic Simulated Annealing Algorithm

B. Finding A Neighbor and Move

Two sets of partitions are called neighbors if we can find a move from one to the other. In this work, we adopt the vertex move, in which a single vertex in one partition, π_i is moved to the other, π_j , such that for an N node graph, the size of the neighborhood for an m -way partition is $S = N \times (m - 1)$. The set of neighbors generated by the vertex move has the advantage of smaller increases in the size of the neighborhood, which is $O(N)$ for a given m , whereas vertex

exchange will give $O(N^2)$ as N increases [13].

C. The Cooling Schedule

The cooling ratio, γ , is a reducing factor for the annealing process getting out of local optima and moving towards a global minimum. To improve the high rejection rate at low temperatures, an *Adaptive Cooling Schedule* was proposed in [15]. One analysis on the adaptive cooling schedule [10] shows longer annealing time always gives better results regardless the design of the cooling schedule. Therefore, we reduce the temperature by a factor of γ per Metropolis iteration for the cooling schedule.

Experiments

The simulations are implemented in C language and conducted on Sun Sparc 1+ stations. For best results in simulated annealing technique, sample experiments were performed for obtaining proper setting of the parameters.

A. Setting the Parameters

There are several parameters in the generic algorithm need to be tuned [10] for simulated annealing: γ , the cooling factor, *SIZEFACTOR* to determine the number of iterations for the Metropolis loop, and *MINPERCENT* as the minimum acceptance ratio for stopping at a low temperature. Here we adopt the analysis from [10] and [13] for the following settings: *SIZEFACTOR*=16, and $\gamma = 0.95$. *MINPERCENT* is set to be 0.005 according to sets of trial simulations.

There is one stopping criterium in our simulation: when the acceptance ratio is below *MINPERCENT* for 5 consecutive temperatures. This is to stop the simulation when most of the trials are rejected.

B. Results and Comparison

A generally accepted set of test cases is the ISCAS 85 benchmarks. Table 2 shows the results for a given input constraint, *CBIT_width*=16. The 'internal net cuts' suggested by the simulated annealing algorithm is for future placement of the modified flip-flops which construct one CBIT dedicated to one partition for testing. These 'cut points' are for pipelined pseudo-exhaustive testing performing distributed test pattern generation (TPG) according to the input number constraint for circuit clusters in a CUT. As outlined in Table 5, pipelined pseudo-exhaustive testing outperforms existing approaches (e.g., PEST [19]) by saving expensive centralized test pattern generator and extra I/O pins/points for controlling/observing the internal cut points. Test quality degradation for pipelined pseudo-exhaustive testing is $O(2^{-N})$ as the number of pipelining stages, k , is much smaller comparing with 2^N .

Table 3 and Table 4 list the previous results from similar approaches in [5] and [6]. Since our partitioning strategy is based on the pipelined testing framework [9], which is different from the testing architecture in [5] and [6], Tables 3 and 4 are for reader's reference only.

Conclusion

We have demonstrated that circuit partitioning for pipelined pseudo-exhaustive testing can be formulated as the NP-complete m -way partition and approached by simulated annealing. Experimental results demonstrate the superiority when simulated annealing is combined with a complete set of cost function and parameter setting.

Future works can be investigated in related areas such as a thorough study on adding a pre-processing program/module to recognize user-specified 'untouchable' cells (or *hard* modules). Extension to sequential circuits with loop cutting and re-timing techniques will give a good modification to current approaches for a better global optimization.

References

- [1] H. Wunderlich and S. Hellerbrand, "The Pseudoexhaustive Test of Sequential Circuits", IEEE Trans. on CAD, vol. 11, no. 1, Jan. 1992, pp. 26-33.
- [2] S. N. Bhatt, F. Chung, and A. L. Rosenberg, "Partitioning Circuits for Improved Testability", Algorithmica, no. 6, pp. 37-48.
- [3] S. Malik, E. M. Sentovich, R. K. Brayton, and A. Sangiovanni-Vincentelli, "Retiming and Resynthesis: Optimizing Sequential Networks with Combinational Techniques", IEEE Trans. on CAD, vol. 10, no. 1, Jan. 1991, pp. 74-84.
- [4] S. C. Seth and V. D. Agrawal, "A New Model for Computation of Probabilistic Testability in Combinational Circuits", Integration, the VLSI Journal, no. 7, 1989, pp. 49-75.
- [5] R. Srinivasan, S. K. Gupta and M. A. Breuer, "An Efficient Partitioning Strategy for Pseudo-Exhaustive Testing", Proc. 1993 DAC, pp. 242-248.
- [6] J. G. Udell, Jr., "Efficient Segmentation for Pseudo-Exhaustive BIST", Proc. 1992 CICC, pp. 13.6.1-5.
- [7] I. Shperling and E. J. McCluskey, "Circuit Segmentation for Pseudo-Exhaustive Testing via Simulated Annealing", Proc. 1987 ITC, pp. 58-66.
- [8] W. B. Jone and C. A. Papachristou, "A Coordinated Approach to Partitioning and Test Pattern Generation for Pseudoexhaustive Testing", Proc. 1989 DAC, pp. 525-530.
- [9] T. Y. Lin and H. Y. Liou, "A New Framework for Designing BIT Multichip Modules with Pipelined Test Strategy", IEEE D&T, vol. 10, no. 4, Dec. 1993, pp. 38-51.
- [10] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning", Operations Research, vol. 37, no. 6, Nov. 1989, pp. 865-892.
- [11] F. Brglez, R. Pownall, and R. Hum, "Accelerated ATPG and Fault Grading via Testability Analysis", Proc. 1985 ISCAS.
- [12] Barry Johnson, "Design and Analysis of Fault Tolerant Digital Systems", Addison-Wesley, Chap. 2 and 6, 1989.
- [13] L. Tao and Y. C. Zhao, "Efficient Heuristic Algorithms for VLSI Circuit Partition", IEE Proc.-G, vol. 140, no. 2, April 1993.
- [14] W. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller, "Equation of State Calculations by Fast Computing Machines", J. Chem. Phys., no. 21, 1953, pp.1087-1092.
- [15] R. Otten and L. van Ginneken, "The Annealing Algorithm", Kluwer Academic Publishers, 1989.
- [16] E. McCluskey and S. Bozorgui-Nesbat, "Design for Autonomous Test", IEEE Trans. on Comp., vol. C-30, no. 11, Nov. 1981, pp. 866-875.
- [17] M. Garey and D. Johnson, "Computer and Intractability-A Guide to the Theory of NP-completeness", W. H. Freeman and Co., Appendix 3.
- [18] J. G. Udell, Jr. and E. J. McCluskey, "An Efficient Segmentation Program for Pseudo-Exhaustive Test", CRC Tech. Rep., no. 88-11, Stanford Univ., Dec. 1988.
- [19] E. Wu, "A Tool for Implementing Pseudo-exhaustive Self-test", AT&T Tech. Journal, Jan. 1991, pp. 87-100.

Circuit Name	Internal nets cut by SA	no. of partitions
C432	49	8
C499	47	8
C880	72	12
C1355	93	28
C1908	203	45
C2670	263	54
C5315	548	124
C6288	1069	164

Table 2: Partition Results from Eq. (1) for $l=16$

Circuit Name	no. of cut points		Internal nets cut by SA
	CSR [5]	ESP-BIST [6]	
C432	27	27	49
C499	8	8	47
C880	11	11	72
C1355	8	8	93
C1908	18	21	203
C2670	33	33	263
C5315	54	54	548
C6288	-	84	1069

Table 3: No. of Cuts Referencing to Related Works for $l=16$

Circuit Name	no. of partitions	
	ESP [18]	SA
C432	-25	8
C499	-40	8
C880	-40	12
C1355	-40	28
C1908	-75	45
C2670	-60	54
C5315	-150	124
C6288	-130	164

Table 4: No. of Partitions Referencing to Related Works for $l=16$

Factor/Test Approaches	PEST [19]	Pipelined Pseudo-exhaustive Testing
Testing Time	$T_{Scan} + T_{MISR} (\approx 2^{max(N_i)} - 1)$	$T_{Scan} + T_{MISR} (\approx 2^{max(N_i)} - 1)$
Area Overhead	$A_{Scan-path} + A_{MISR}$	$min(A_{Scan-path}) + A_{MISR}$
Extra TPG Circuitry	Extra Boundary Scan Circuitry	NONE
Fault Coverage/Test Effectiveness	$1 - 2^{-N}$	$1 - k \times 2^{-N}$ from [9]

Table 5: Comparison between PEST Approach and Pipelined Signature Analysis

Performability Analysis of Non-repairable Multicomponent Systems Using Order Statistics *

Amiya Bhattacharya
CSE Department
(619)534-7883
amiya@cs.ucsd.edu

Ramesh R. Rao
ECE Department
(619)534-6433
rrao@ucsd.edu

Ting-Ting Y. Lin
ECE Department
(619)534-4738
lin@ece.ucsd.edu

University of California, San Diego
La Jolla, CA 92093

Abstract

Performability, a composite measure that integrates both performance and reliability, has been deemed to be essential in evaluating systems that are capable of trading off performance for reliability under component failures. For non-repairable systems, the goal is to evaluate the distribution or moments of some accumulated reward (performance) defined on a stochastic process that characterizes the different configurations, as successive faults appear. In this paper, instead of assuming that the distributions of sojourn time at various configurations are known, we deduce them from the distributions of the individual component lifetime. We allow lifetimes to be arbitrary but assume that they are independent from one component to another. Knowledge of the reward rates in different configurations allows us to relate the sojourn times to the order statistics of the component lifetimes. We then solve for the distributions of the order statistics. The approach is illustrated in context of a parallel system made of identical components.

Keywords: Graceful degradation, Reliability, Performance, Sojourn time.

1 Introduction

Stochastic modelling and analysis have been extensively used in quantitative evaluation of both the reliability and performance of computer or communication systems. Traditionally, they have been studied as two separate aspects. However, a need for a composite metric for integrating these two arises for analysing the so-called *degradable systems*. By virtue of their fault detection and reconfiguration capability, they can operate in several degraded modes as a consequence of component failures.

Degradable systems form a special class of fault-tolerant systems where reliability can be traded off at the cost of performance. Performability, a measure that combines both performance and reliability, has been developed as an accepted standard to capture this trade-off [2, 17]. It is defined as the probability that the system reaches an accomplishment level over a utilization interval called the *mission time*. As faults occur, the system goes through different configurations that can be characterized by a stochastic process. Oftentimes, even if the system is potentially repairable, the designer is concerned with the transient behavior of the system over the mission. Thus the problem reduces to the evaluation of the distribution and/or moments of some accumulated performance metric (reward) defined on the stochastic process. This freedom of choice for the reward makes the measure more powerful and versatile.

The first effort at integrating performance and reliability for a degradable computing system was made by Beaudry [1], where the total amount of computation available from a degrading computer system was

*Supported by ONR under grant N0001491-J1017 and NSF under grant NCR-8904029

analyzed under a Markov model of processor failure. In another early work, Huslende [4] proposed a different measure, the probability that the system performance remains above a certain level at all time during a mission. In the framework of performability introduced by Meyer [2], it was clear that both these measures (and many others) could be interpreted as different assignment of reward rates [8].

Meyer [5] and Furchtgott and Meyer [6] showed how to obtain closed form solution to the performability distribution by conditioning on the state trajectories (sample paths). We closely follow the outline for analysing systems with arbitrary stochastics of failure that was presented in [6]. Before we present our approach, it should be noted that the integral solutions which arise require enumeration of all possible state trajectories involving numerical algorithms with linear complexity in terms of the trajectories. For a large system, the computations involved could be very complex depending on the degradation pattern. A semi-Markovian characterization of the underlying stochastic process was considered appropriate for a non-repairable degrading system with independent failures [9], and found to lead to tractable solution [6]. However, as illustrated by Ciardo et al. [12], identifying the sojourn time distribution in the semi-Markov model may not be straightforward for a system with complex configuration.

Due to this complexity, most researchers analyzed performability using the Markov reward model [5, 7, 8, 9, 10, 11, 14, 19], where the problem reduces to the transient analysis of the underlying Markov chain. This was basically a generalization of Beaudry's idea captured in the new framework of performability. However, due to the memoryless property of Markov process, it implies an exponential distribution of sojourn time, and a constant hazard rate. In practice, system components have been better characterized by varying hazard rate over the lifetime, often modelled by Weibull distribution [18]. Although use of exponential approximation to non-exponential distribution may be sufficient for a steady-state analysis [18], it can introduce considerable error in transient analysis [12].

The definition of performability we use is derived from Meyer [2]. The system we consider here has been studied earlier in [5] and [8], where the underlying stochastic process was assumed to be Markov, giving rise to the memoryless distribution of sojourn times at all degraded configurations. The method outlined in this paper is a more general in this regard. We deduce the distributions of the sojourn time from the distributions of individual component lifetimes, that are assumed to be independent from one component to another. The lifetimes need not be memoryless; they can be arbitrary. A semi-Markovian process automatically arises out of the order statistic formulation, but there is no need for explicitly defining its sojourn time distribution.

The rest of the paper is organized as follows. The definition of performability and related measures is presented in section 2, along with the related assumptions and an example. In section 3, we formulate the performability problem for a system with no critical component in terms of the order statistics of random variables representing component lifetimes, and derive closed form solutions for the related measures. These results are then extended to the system with critical component in section 4. In section 5, some numerical values for the derived expressions are plotted to illustrate the effect of the critical component and its mean lifetime. We conclude the paper in section 6 by summarizing the contribution with a focus to further work.

2 Overview

Let $\{X_t\}_{t \geq 0}$ be a continuous-time discrete-space stochastic process representing the state of the degrading system. The state-space Q is finite for any practical system, however large it may be. Each state $q \in Q$ represents a possible configuration of the system with some components failed, and has an associated *reward rate* $\rho_q = \rho(q)$, a non-negative real number reflecting the level of performance per unit time, when the system is in state q . The system we analyze here is made of n identical components, and it degrades gracefully when the components fail one by one. The states are $q_k, k = 0, \dots, n$, where q_k denotes the state with k faulty components.

Performability of the system over a *mission time* t is defined as the *probability density function* of the accumulated reward (performance)

$$Y_t = \int_0^t \rho(X_s) ds. \quad (1)$$

The *cumulative distribution function* of Y_t is often referred to as the *performability distribution function*. The mean value of Y_t , which we would call *mean reward before failure* (MRBF), is a useful characterization.

For a nonrepairable system, the transitions between configurations take the shape of an acyclic graph, so that reentry to a state is not feasible. The *sojourn time* or the *residence time* τ_q , which is the total time spent by the system in configuration q over the mission time $t = \sum_{q \in Q} \tau_q$, is therefore contiguous. Equation (1) thus reduces to

$$Y_t = \sum_{q \in Q} \rho_q \tau_q, \quad (2)$$

which also provides a simple expression for MRBF

$$E[Y_t] = \sum_{q \in Q} \rho_q E[\tau_q]. \quad (3)$$

When the accumulated reward until the complete system failure is of interest, one should consider the value of Y_t as $t \rightarrow \infty$. The corresponding measures to be considered are the distribution functions and mean value of

$$Y = \lim_{t \rightarrow \infty} Y_t = \int_0^\infty \rho(X_t) dt, \quad (4)$$

which we henceforth consider in this paper.

Example Let us consider a single component system having just the working and failed states. If we assign $\rho = 1$ for the fault-free state and $\rho = 0$ for the failed one, the performability distribution function reduces to

$$\Pr[Y \leq y] = \Pr[T \leq y],$$

where T is the random variable representing the lifetime of the component. The reliability $R(t)$ can be written as

$$\begin{aligned} R(t) &= \Pr[T > t] \\ &= 1 - \Pr[T \leq t] \\ &= 1 - \Pr[Y \leq t]. \end{aligned}$$

Clearly, the MRBF reduces to the mean time to failure (MTTF) [18]. \square

One implicit assumption behind formulating a composite performance-reliability measure as above is that the performance and failure rates are functions of only the system state. This means that increasing load or stress during the degradation due to component failure does not affect the lifetime distribution of the surviving components. This is not only a feature which makes the analysis tractable, but also a desired characteristic in many implementations.

3 Order Statistics Formulation of Performability without Critical Component

In the previous section we identified the state-space associated with the graceful degradation of an n -component system. We start this section with the assumption that all the states $q_k, k = 0, \dots, n$ are visited in that strict order as the system degrades gracefully. In effect, we ignore the possibility of any abrupt failure caused by the failure of any critical component that holds the n components together. The boundaries of the sojourn times can now be identified with the time of failure of the successive components. These are precisely the order statistics of the random variables representing lifetime of the components. In subsection 3.1, we formally introduce the order statistics, and explore how their distributions relate to the distributions of the component lifetimes. Subsection 3.2 shows the result from a traditional Markov model of the order statistics variables to be a special case of these formulations. Finally in subsection 3.3, the closed form expressions for the performability distribution and MRBF are derived in terms of the distributions of the order statistics of component lifetimes.

3.1 Order statistics of lifetime

If a family of random variables T_1, T_2, \dots, T_n are arranged in ascending order of magnitude and renamed as $T_{1:n} \leq T_{2:n} \leq \dots \leq T_{n:n}$, the new random variable $T_{k:n}$ is called the k th order statistic for $k = 1, 2, \dots, n$. Suppose T_1, T_2, \dots, T_n represent the lifetimes of the identical components, and as a result, they are iid (independent and identically distributed) with cdf $F(t) = \Pr[T_i \leq t]$. The cdf $F_{k:n}(t)$ of the order statistics $T_{k:n}$ of lifetimes can be derived as

$$\begin{aligned} F_{k:n}(t) &= \Pr[T_{k:n} \leq t] \\ &= \Pr[k \text{ or more components fail in } [0, t]] \\ &= \sum_{i=k}^n \Pr[\text{Exactly } i \text{ components fail in } [0, t]] \\ &= \sum_{i=k}^n \binom{n}{i} \{F(t)\}^i \{1 - F(t)\}^{n-i}, \end{aligned} \quad (5)$$

which is the tail probability of a binomial distribution. Using an identity (see appendix for details), This can also be written as an incomplete beta function,

$$\begin{aligned} F_{k:n}(t) &= I_{F(t)}(k, n - k + 1) \\ &= \frac{1}{B(k, n - k + 1)} \int_0^{F(t)} u^{k-1} (1 - u)^{n-k} du. \end{aligned} \quad (6)$$

To derive the pdf $f_{k:n}(t)$, let us consider the probability of the event that the k th failure occurs in $[t, t + \delta t]$. This means $k - 1$ components must fail in $[0, t]$, and $n - k$ must fail in $[t + \delta t, \infty)$. Moreover, there are $n!/(k-1)!(n-k)!$ ways to choose the first $k-1$, the k th and the rest $n-k$ components out of n . As all components fail independently, we have

$$\begin{aligned} \Pr[t \leq T_{k:n} \leq t + \delta t] &= \frac{n!}{(k-1)!(n-k)!} \{F(t)\}^{k-1} f(t) \delta t \{1 - F(t + \delta t)\}^{n-k}. \end{aligned}$$

Dividing both sides by δt and taking limit as $\delta t \rightarrow 0$, we get

$$f_{k:n}(t) = \frac{n!}{(k-1)!(n-k)!} \{F(t)\}^{k-1} \{1 - F(t)\}^{n-k} f(t) \quad (7)$$

The same expression for $f_{k:n}(t)$ can also be obtained by differentiating $F_{k:n}(t)$ in equation (6) with respect to t .

A straightforward extension of the above argument yields an expression for the joint pdf of two or more order statistics of the lifetime distribution of individual components. For the pairwise joint pdf $f_{i,j:n}(s, t)$, $s \leq t$, we consider the event that the i th failure occurs in $[s, s + \delta s]$ and j th failure occurs in $[t, t + \delta t]$. The first $i-1$ failures occur in $[0, s]$, $j-i-1$ intermediate failures occur in $[s + \delta s, t]$ and the remaining $n-j$ failures occur in $[t + \delta t, \infty)$. The choice of the sequence can be done in $n!/(i-1)!(j-i-1)!(n-j)!$ ways. Due to the independence of their lifetime

$$\begin{aligned} \Pr[s \leq T_{i:n} \leq s + \delta s, t \leq T_{j:n} \leq t + \delta t] &= \frac{n!}{(i-1)!(j-i-1)!(n-j)!} \{F(s)\}^{i-1} f(s) \delta s \\ &\quad \{F(t) - F(s + \delta s)\}^{j-i-1} f(t) \delta t \{1 - F(t + \delta t)\}^{n-j} \end{aligned}$$

which gives

$$\begin{aligned} f_{i,j:n}(s, t) &= \frac{n!}{(i-1)!(j-i-1)!(n-j)!} \{F(s)\}^{i-1} \{F(t) - F(s)\}^{j-i-1} \\ &\quad \{1 - F(t)\}^{n-j} f(s) f(t) \end{aligned} \quad (8)$$

By similar reasoning, we can have a general expression for the joint pdf of any k order statistics [3]. For $1 \leq n_1 < n_2 < \dots < n_k \leq n$ and $t_1 \leq t_2 \leq \dots \leq t_k$,

$$f_{n_1, \dots, n_k}(t_1, \dots, t_k) = \frac{n!}{(n_1 - 1)!(n_2 - n_1 - 1)! \dots (n - n_k)!} \{F(t_1)\}^{n_1-1} f(t_1) \{F(t_2) - F(t_1)\}^{n_2-n_1-1} f(t_2) \dots \{1 - F(t_k)\}^{n-n_k} \quad (9)$$

3.2 Comparison with a Markov model

Before evaluating performability or MRBF, let us verify that we have indeed arrived at a more general solution of sojourn time boundaries than what we could expect by transient analysis of Markov model of the system. With Markovian assumptions, the chain of transitions translates to a *pure death process* with a constant hazard rate λ . At most one transition due to failure can occur in the small interval $[t, t + \delta t]$. This gives rise to exponentially distributed component lifetime with hazard rate λ [18],

$$F(t) = 1 - e^{-\lambda t} \quad (10)$$

The transition probabilities are given by

$$\begin{aligned} P_{k,k+1} &= (n-k)\lambda \delta t, & k = 0, \dots, n-1 \\ P_{k,k} &= 1 - (n-k)\lambda \delta t, & k = 0, \dots, n-1 \\ P_{n,n} &= 1 \\ P_{i,j} &= 0 & \text{otherwise} \end{aligned} \quad (11)$$

Denoting $\Pr[T_{k:n} \leq t]$ by $p_k(t)$, for $k = 1, \dots, n$, we have

$$\begin{aligned} p_k(t + \delta t) &= \Pr[T_{k:n} \leq t + \delta t] \\ &= \Pr[k \text{ or more components fail in } [0, t + \delta t]] \\ &= 1 \times \Pr[k \text{ or more components fail in } [0, t]] + \\ &\quad \Pr[\text{One failure in } [t, t + \delta t] \mid \text{Exactly } k-1 \text{ components fail in } [0, t]] \\ &\quad \times \Pr[\text{Exactly } k-1 \text{ components fail in } [0, t]] \\ &= \Pr[T_{k:n} \leq t] + \{\Pr[T_{k-1:n} \leq t] - \Pr[T_{k:n} \leq t]\} (n-k+1) \lambda \delta t \\ &= p_k(t) + \{p_{k-1}(t) - p_k(t)\} (n-k+1) \lambda \delta t \end{aligned}$$

from which we obtain

$$\begin{aligned} p'_k(t) &= \lim_{\delta t \rightarrow 0} \frac{p_k(t + \delta t) - p_k(t)}{\delta t} \\ &= (n-k+1) \{p_{k-1}(t) - p_k(t)\} \end{aligned} \quad (12)$$

with $p_0(0) = 1$ and $p_k(0) = 0, k \neq 0$ as the initial conditions.

Our formulation of section 3.1 is of course more general in that it can accommodate arbitrary distribution. We now show that by setting $F(t) = 1 - e^{-\lambda t}$ equation (12) can be derived from equations (5) and (6). Differentiating the incomplete beta function form of equation (6)

$$p_k(t) = \frac{1}{B(k-1, n-k+1)} \int_0^{1-e^{-\lambda t}} u^{k-1} (1-u)^{n-k} du$$

we get

$$p'_k(t) = \frac{n!}{(k-1)!(n-k)!} \lambda e^{-(n-k+1)\lambda t} (1 - e^{-\lambda t})^{k-1},$$

while using the binomial tail probability form of equation (5)

$$p_k(t) = \sum_{i=k}^n \binom{n}{i} (1 - e^{-\lambda t})^i (e^{-\lambda t})^{n-i}$$

we get

$$\begin{aligned} & (n-k+1)\lambda \{p_{k-1}(t) - p_k(t)\} \\ &= (n-k+1)\lambda \left\{ \sum_{i=k-1}^n \binom{n}{i} (1 - e^{-\lambda t})^i (e^{-\lambda t})^{n-i} \right. \\ & \quad \left. - \sum_{i=k}^n \binom{n}{i} (1 - e^{-\lambda t})^i (e^{-\lambda t})^{n-i} \right\} \\ &= (n-k+1)\lambda \binom{n}{k-1} (1 - e^{-\lambda t})^{k-1} (e^{-\lambda t})^{n-k+1} \\ &= \frac{n!}{(k-1)!(n-k)!} \lambda e^{-(n-k+1)\lambda t} (1 - e^{-\lambda t})^{k-1}, \end{aligned}$$

verifying that $p'_k(t) = (n-k+1)\lambda \{p_{k-1}(t) - p_k(t)\}$.

3.3 Evaluation of performability

We are now ready to evaluate the performability of the n -component parallel system. Clearly $T_{k:n}, k = 1, \dots, n$ are the sojourn time boundaries. Defining $T_{0:n} = 0$, the sojourn time τ_k at state q_k is

$$\tau_k = T_{k+1:n} - T_{k:n}, \quad k = 0, \dots, n-1 \quad (13)$$

The accumulated reward in equation (4) can now be written as (see figure 1)

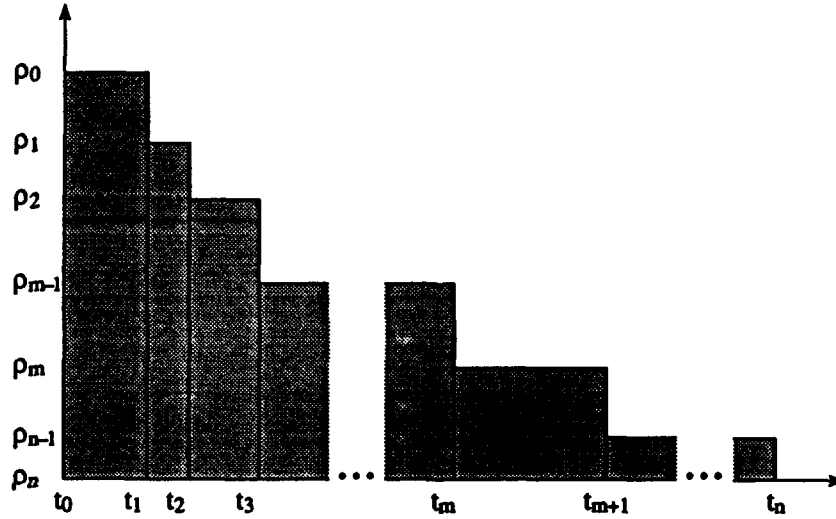


Figure 1: Accumulated reward without critical component

$$\begin{aligned}
Y &= \sum_{k=0}^{n-1} \rho_k \tau_k \\
&= \sum_{k=0}^{n-1} \rho_k (T_{k+1:n} - T_{k:n}), \quad T_{0:n} \triangleq 0 \\
&= \sum_{k=1}^n (\rho_{k-1} - \rho_k) T_{k:n}, \quad \rho_n \triangleq 0
\end{aligned} \tag{14}$$

so that

$$\begin{aligned}
\Pr[Y \leq y] &= \Pr\left[\sum_{k=1}^n (\rho_{k-1} - \rho_k) T_{k:n} \leq y\right] \\
&= \int \cdots \int_{\sum_{k=1}^n (\rho_{k-1} - \rho_k) t_k \leq y} f_{1,\dots,n:n}(t_1, \dots, t_n) dt_1 \cdots dt_n
\end{aligned} \tag{15}$$

Similarly, the MRBF reduces to

$$\begin{aligned}
E[Y] &= E\left[\sum_{k=1}^n (\rho_{k-1} - \rho_k) T_{k:n}\right] \\
&= \sum_{k=1}^n (\rho_{k-1} - \rho_k) E[T_{k:n}] \\
&= \sum_{k=1}^n (\rho_{k-1} - \rho_k) \int_{t=0}^{\infty} t f_{k:n}(t) dt
\end{aligned} \tag{16}$$

4 Effect of Critical Components

Until now we have not considered system failure due to failure of any critical component. The effect of such a component failure is an abrupt system failure — the graceful degradation gets truncated at that point. This phenomenon is very common in practical systems. However, the results derived in the previous section would still apply to systems where the critical component has much higher reliability as compared to the failing components.

To incorporate the effect of critical components, we have to change the expression for the sojourn time by

$$\begin{aligned}
\tau_k &= 0, & T_c < T_{k:n} \\
&= T_c - T_{k:n}, & T_{k:n} \leq T_c \leq T_{k+1:n} \\
&= T_{k+1:n} - T_{k:n}, & T_{k+1:n} \leq T_c
\end{aligned} \tag{17}$$

where T_c denotes the lifetime of the critical component, distributed with cdf $F_c(t)$ and pdf $f_c(t)$.

It is also quite straightforward to handle multiple critical components. One can lump the effect onto a single critical component of effective lifetime T_c with cdf $F_c(t)$, by considering the hypothetical critical component to be active until the first real critical component failure. Assuming there are n_c critical components with individual lifetimes $T_{c_1}, T_{c_2}, \dots, T_{c_{n_c}}$ with cdf $F_{c_1}(t), F_{c_2}(t), \dots, F_{c_{n_c}}(t)$ (they may as well be different), the effective distribution of critical component lifetime is given by

$$F_c(t) = 1 - \prod_{i=1}^{n_c} \{1 - F_{c_i}(t)\}, \tag{18}$$

and $f_c(t)$ can be obtained by differentiating $F_c(t)$ with respect to time.

Given that $T_{m:n} < T_c \leq T_{m+1:n}$, Y from equation (4) reduces to (see figure 2)

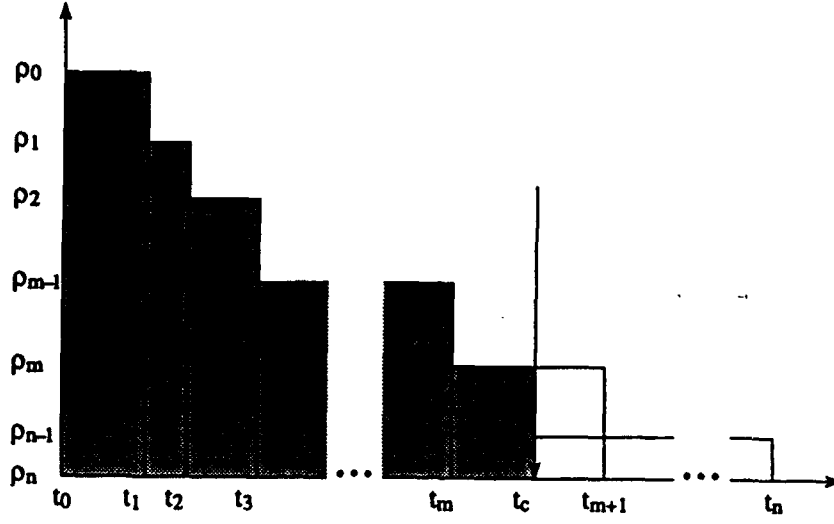


Figure 2: Accumulated reward with critical component

$$\begin{aligned}
 Y &= \sum_{k=0}^m \rho_k \tau_k \\
 &= \sum_{k=0}^{m-1} \rho_k (T_{k+1:n} - T_{k:n}) + \rho_m (T_c - T_{m:n}) \\
 &= \sum_{k=1}^m (\rho_{k-1} - \rho_k) T_{k:n} + \rho_m T_c.
 \end{aligned} \tag{19}$$

As $T_{m:n} < T_c \leq T_{m+1:n}$ are disjoint, by the axiom of probability

$$\begin{aligned}
 \Pr[Y \leq y] &= \Pr[\{Y \leq y\} \cap \{T_c \leq T_{1:n}\}] \\
 &\quad + \sum_{m=1}^{n-1} \Pr[\{Y \leq y\} \cap \{T_{m:n} < T_c \leq T_{m+1:n}\}] \\
 &\quad + \Pr[\{Y \leq y\} \cap \{T_n < T_c\}] \\
 &= \Pr[\{\rho_0 T_c \leq y\} \cap \{T_c \leq T_{1:n}\}] \\
 &\quad + \sum_{m=1}^{n-1} \Pr[\sum_{k=1}^m \{(\rho_{k-1} - \rho_k) T_{k:n} + \rho_m T_c \leq y\} \cap \{T_{m:n} < T_c \leq T_{m+1:n}\}] \\
 &\quad + \Pr[\sum_{k=1}^n \{(\rho_{k-1} - \rho_k) T_{k:n} \leq y\} \cap \{T_{n:n} < T_c\}]
 \end{aligned}$$

$$\begin{aligned}
&= \int \int_{\rho_0 t_c \leq y, t_c \leq t_1} f_{1:n}(t_1) f_c(t_c) dt_1 dt_c \\
&+ \sum_{m=1}^{n-1} \left\{ \int \cdots \int_{\sum_{k=1}^m (\rho_{k-1} - \rho_k) t_k + \rho_m t_c \leq y, t_m < t_c \leq t_{m+1}} f_{1,\dots,m+1:n}(t_1, \dots, t_{m+1}) f_c(t_c) dt_1 \cdots dt_{m+1} dt_c \right\} \\
&+ \int \cdots \int_{\sum_{k=1}^n (\rho_{k-1} - \rho_k) t_k \leq y, t_n < t_c} f_{1,\dots,n:n}(t_1, \dots, t_n) f_c(t_c) dt_1 \cdots dt_n dt_c
\end{aligned} \tag{20}$$

An expression for MRBF can be obtained by conditioning on the same set of disjoint events as

$$\begin{aligned}
E[Y] &= E[Y | T_c \leq T_{1:n}] \cdot \Pr[T_c \leq T_{1:n}] \\
&+ \sum_{m=1}^{n-1} E[Y | T_{m:n} < T_c \leq T_{m+1:n}] \cdot \Pr[T_{m:n} < T_c \leq T_{m+1:n}] \\
&+ E[Y | T_{n:n} < T_c] \cdot \Pr[T_{n:n} < T_c] \\
&= E[\rho_0 T_c | T_c \leq T_{1:n}] \cdot \Pr[T_c \leq T_{1:n}] \\
&+ \sum_{m=1}^{n-1} E[\sum_{k=1}^m (\rho_{k-1} - \rho_k) T_{k:n} + \rho_m T_c | T_{m:n} < T_c \leq T_{m+1:n}] \\
&\quad \cdot \Pr[T_{m:n} < T_c \leq T_{m+1:n}] \\
&+ E[\sum_{k=1}^n (\rho_{k-1} - \rho_k) T_{k:n} | T_{n:n} < T_c] \cdot \Pr[T_{n:n} < T_c] \\
&= \rho_0 \int \int_{t_c \leq t_1} t_c f_{1:n}(t_1) f_c(t_c) dt_1 dt_c \\
&+ \sum_{m=1}^{n-1} \left\{ \sum_{k=1}^{m-1} (\rho_{k-1} - \rho_k) \int \int \int_{t_k \leq t_m < t_c \leq t_{m+1}} t_k f_{k,m,m+1:n}(t_k, t_m, t_{m+1}) f_c(t_c) dt_k dt_m dt_{m+1} dt_c \right. \\
&\quad + (\rho_{m-1} - \rho_m) \int \int \int_{t_m < t_c \leq t_{m+1}} t_m f_{m,m+1:n}(t_m, t_{m+1}) f_c(t_c) dt_m dt_{m+1} dt_c \\
&\quad \left. + \rho_m \int \int \int_{t_m < t_c \leq t_{m+1}} t_c f_{m,m+1:n} f_c(t_c) dt_m dt_{m+1} dt_c \right\} \\
&+ \sum_{k=1}^{n-1} (\rho_{k-1} - \rho_k) \int \int \int_{t_k \leq t_n < t_c} t_k f_{k,n:n}(t_k, t_n) f_c(t_c) dt_k dt_n dt_c \\
&+ (\rho_{n-1} - \rho_n) \int \int_{t_n < t_c} t_n f_{n:n}(t_n) f_c(t_c) dt_n dt_c.
\end{aligned} \tag{21}$$

5 Numerical Evaluation and Results

To illustrate how the formulations developed here can be used for computing performability distribution and MRBF, we consider a perfectly scalable system made of n components and a single critical component. In

other words, this indicates a reward assignment of the form

$$\rho_k = (n - k)\rho, \quad k = 0, \dots, n$$

The analysis only suggests that the sequence $\rho_0, \rho_1, \dots, \rho_n$ is monotonically nonincreasing, they could depend on the system states q_0, q_1, \dots, q_n in a potentially complex way. On the other hand, a scalable reward assignment can be used as a consistency check on the correctness of numerical routines, by verifying whether or not linearity between number of components and MRBF results.

To obtain the numerical values, the series of integrals in equations (15–16) and (20–21) need be evaluated. We use multidimensional Gaussian quadrature to perform the integrations numerically in the specified regions. For illustrations and comparisons, we assume that the lifetimes of both the critical and non-critical components are modelled by Weibull distribution,

$$\Pr[T \leq t] = 1 - e^{-(\lambda t)^\alpha},$$

the parameters λ and α being different in general. The necessary marginal and joint density functions of the order statistics can then be used to compute the values of the integrands at any point. The grid points for the quadrature are generated by recursively using Gauss-Legendre formula [16], with a suitably large value chosen for infinity. This was found to be numerically stable and well-behaved for the exponential class of integrands, as compared to a combination of Gauss-Laguerre and Gauss-Legendre quadrature designed for semi-open regions.

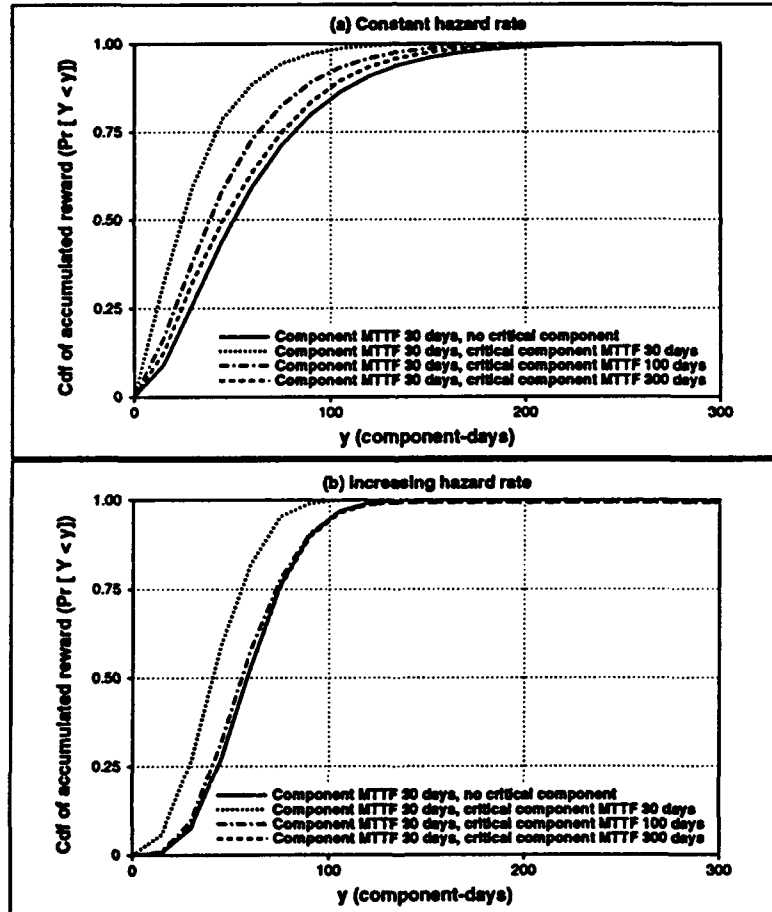


Figure 3: Performability distribution ($\#components = 2$, $\alpha = 1$ vs. 2)

Figure 3 shows the effect of critical component on the performability distribution function for a two component system, with $\alpha = 1$ and 2 in (a) and (b) respectively. We restrict the number of components to

$n = 2$, because for higher values, long CPU hours are needed to evaluate the related integrals of upto $(n + 1)$ dimensions to the desired accuracy. The unit used for the accumulated reward is *component-day*, the total throughput available from a component over one day. As the critical components need be more reliable than the non-critical components, we increase the MTTF of the critical component keeping the MTTF of the non-critical components unchanged. The cdf of the accumulated reward of the system, as a result, approaches in limit that of a system with no critical components. The parameter $\alpha = 1$ models the Markovian system with constant hazard rate, whereas $\alpha = 2$ corresponds to a case of increasing hazard rate. The motivation behind studying systems with the increasing hazard rate is that Markovian approximation for them may lead to over-optimistic results. With increasing MTTF of critical component, the performability distribution approaches the limiting case faster for the increasing hazard rate.

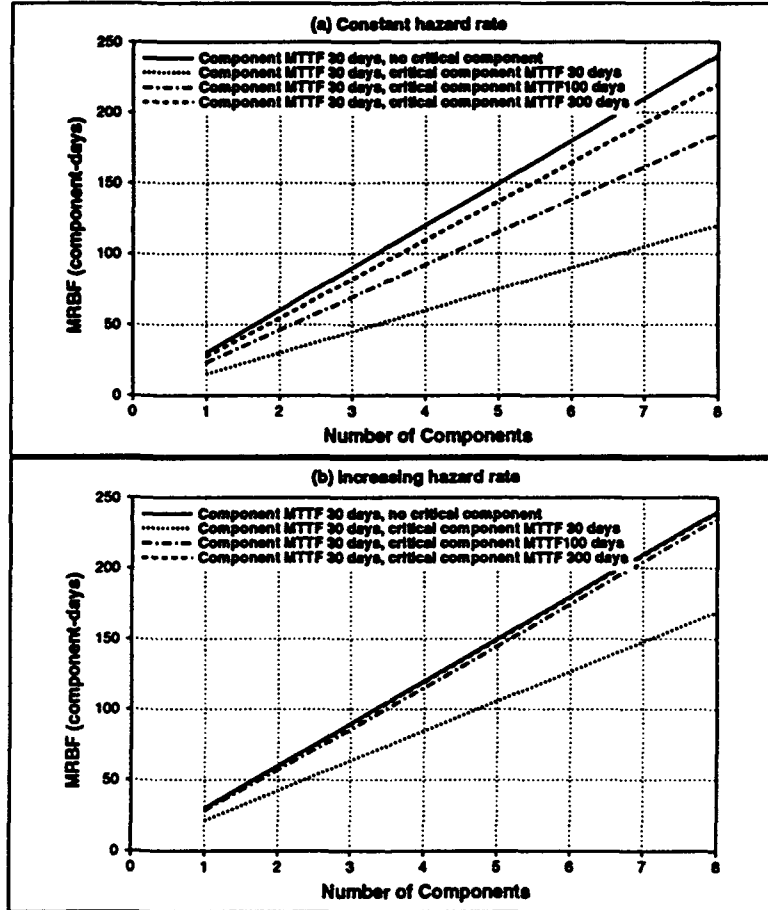


Figure 4: Mean reward before failure ($\alpha = 1$ vs. 2)

The observations are, however, not limited to systems with two components. Computing MRBF for systems with critical components requires evaluation of only upto 4-dimensional integrals, although the total number of such integral evaluations grows as $O(n^2)$. As a result, computing MRBF for larger system is still possible whereas computing performability distribution is not. We have computed the MRBFs of systems with upto eight identical components, and they have been plotted against the number of components in figure 4 for $\alpha = 1$ and 2 in (a) and (b) respectively. Naturally, component-day is also the choice of unit for MRBF. With increasing MTTF of critical component, the MRBF of the system approaches the limiting MRBF of the system with no critical component. As we have observed in the case of performability distribution, this is more prominent with increasing hazard rate as compared to the Markovian or constant hazard rate case. Clearly MRBF is a good indicator of performability in that it preserves the trend. This also means that the reliability requirement of the critical component to achieve the best possible performability

is less stringent as the hazard rate increases.

6 Conclusions

The usefulness of a closed form analytical solution for any system measure cannot be overemphasized. In this paper, we have developed a methodology based on order statistics to analytically evaluate performability figures for a non-repairable multicomponent parallel system with one or more critical elements. This description indeed characterizes a simple yet broad class of parallel systems. An SIMD computer can be cited as one example where the processing elements (PEs) form the pool of degradable components, whereas the master CPU, the controller and the clock driver can be identified as the critical components. The applicability of the approach is, however, not limited only to the configuration mentioned above. Under suitable bounding arguments, similar degradation steps can be found embedded in more complex parallel processing systems. The scope of extending the formulation exists in some arrays and interconnection networks.

Our approach breaks the tradition in that it does not call for making assumptions about the sojourn time distributions, whether memoryless or otherwise. The key assumption is that of the independence of component lifetime, which is more tangible to a system designer. Lifetime of a component can have any general (known) distribution, possibly supplied by the manufacturer. This is clearly in contrast to the Markov model, where hazard rates for components are assumed constant. Thus, in a real system with increasing hazard rate, Markovian analysis would come up with over-optimistic estimates, whereas our analysis should be more accurate. The technique can also prove useful to study the effect of the critical component and to do a requirement analysis on its MTTF.

A Appendix: Incomplete Beta function

We have used an identity relating the partial binomial sum to the incomplete beta function. For $0 \leq p \leq 1$, $a > 0$, $b > 0$, the incomplete beta function $I_p(a, b)$ is defined to be

$$I_p(a, b) = \frac{1}{B(a, b)} \int_0^p t^{a-1} (1-t)^{b-1} dt$$

where

$$B(a, b) = \int_0^1 t^{a-1} (1-t)^{b-1} dt$$

is the complete beta function. A proof for the identity

$$I_p(k, n-k+1) = \sum_{i=k}^n \binom{n}{i} p^i (1-p)^{n-i},$$

for integer $n \geq k$, is given here.

Proof: Integrating LHS by parts, we have

$$\begin{aligned} I_p(k, n-k+1) &= \frac{1}{B(k, n-k+1)} \int_0^p t^{k-1} (1-t)^{n-k} dt \\ &= \frac{1}{B(k, n-k+1)} \left\{ \left[\frac{t^k}{k} (1-t)^{n-k} \right]_0^p + \int_0^p \frac{t^k}{k} (n-k) (1-t)^{n-k-1} dt \right\} \\ &= \frac{\Gamma(n+1)}{\Gamma k \Gamma(n-k+1)} \left\{ \frac{1}{k} p^k \{1-p\}^{n-k} + \frac{n-k}{k} \int_0^p t^k (1-t)^{n-k-1} dt \right\} \\ &= \frac{n!}{k!(n-k)!} p^k \{1-p\}^{n-k} + \frac{\Gamma(n+1)}{\Gamma(k+1) \Gamma(n-k)} \int_0^p t^k (1-t)^{n-k-1} dt \end{aligned}$$

$$\begin{aligned}
&= \binom{n}{k} p^k \{1-p\}^{n-k} + \frac{1}{B(k+1, n-k)} \int_0^p t^k (1-t)^{n-k-1} dt \\
&= \binom{n}{k} p^k \{1-p\}^{n-k} + I_p(k+1, n-k)
\end{aligned}$$

Observing that

$$\begin{aligned}
I_p(n, 1) &= \frac{1}{B(n, 1)} \int_0^p t^{n-1} dt \\
&= \frac{\Gamma(n+1)}{\Gamma n \Gamma 1} \left[\frac{t^n}{n} \right]_0^p \\
&= p^n
\end{aligned}$$

The recurrence can be solved as

$$\begin{aligned}
&I_p(k, n-k+1) \\
&= I_p(k+1, n-k) + \binom{n}{k} p^k (1-p)^{n-k} \\
&= I_p(k+2, n-k-1) + \binom{n}{k} p^k (1-p)^{n-k} + \binom{n}{k+1} p^{k+1} (1-p)^{n-k-1} \\
&\vdots \\
&= I_p(n, 1) + \sum_{i=k}^{n-1} \binom{n}{i} p^i (1-p)^{n-i} \\
&= \sum_{i=k}^n \binom{n}{i} p^i (1-p)^{n-i}. \quad \square
\end{aligned}$$

References

- [1] M. D. Beaudry, "Performance-related reliability measures for computing systems." *IEEE Transactions on Computers*, vol. 27, pp. 540-547, June 1978.
- [2] J. F. Meyer, "On evaluating the performability of degradable computing systems." *IEEE Transactions on Computers*, vol. 29, pp. 720-731, August 1980.
- [3] H. A. David, *Order Statistics*, 2nd ed., John Wiley & Sons, 1981.
- [4] R. Huslende, "A combined evaluation of performance and reliability for degradable systems." *Proc. ACM SIGMETRICS*, pp. 157-164, September 1981.
- [5] J. F. Meyer, "Closed-form solution of performability." *IEEE Transactions on Computers*, vol. 31, pp. 648-657, July 1982.
- [6] D. G. Furchtgott and J. F. Meyer, "A performability solution method for degradable nonrepairable systems." *IEEE Transactions on Computers*, vol. 33, pp. 550-554, June 1984.
- [7] B. R. Iyer, L. Donatiello and P. Heidelberger, "Analysis of performability for stochastic models of fault-tolerant systems." *IEEE Transactions on Computers*, vol. 35, pp. 902-907, October 1986.
- [8] L. Donatiello and B. R. Iyer, "Analysis of a composite performance reliability measure for fault-tolerant systems." *Journal of the ACM*, vol. 34, pp. 179-189, January 1987.
- [9] A. Goyal and A. N. Tantawi, "Evaluation of performability for degradable computer systems." *IEEE Transactions on Computers*, vol. 36, pp. 738-744, June 1987.

- [10] R. M. Smith, K. S. Trivedi and A. V. Ramesh, "Performability analysis: measures, an algorithm and a case study." *IEEE Transactions on Computers*, vol. 37, pp. 406-417, April 1988.
- [11] K. R. Pattipati and S. S. Shah, "On the computational aspects of performability models of fault-tolerant computer systems," *IEEE Transactions on Computers*, vol. 39, pp. 832-836, June 1990.
- [12] G. Ciardo, R. Marie, B. Sericola and K. S. Trivedi, "Performability analysis using semi-Markov reward processes." *IEEE Transactions on Computers*, vol. 39, pp. 1251-1264, October 1990.
- [13] W. H. Sanders and J. F. Meyer, "Reduced base model construction methods for stochastic activity networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 25-36, January 1991.
- [14] L. Donatiello and V. Grassi, "On evaluating the cumulative performance distribution of fault-tolerant computer systems," *IEEE Transaction on Computers*, vol. 40, pp. 1301-1307, November 1991.
- [15] B. C. Arnold, N. Balakrishnan and H. N. Nagaraja, *A First Course in Order Statistics*, John Wiley & Sons, 1992.
- [16] W. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed., Cambridge University Press, 1992.
- [17] J. F. Meyer, "Performability: a retrospective and some pointers to the future," *Performance Evaluation*, vol. 14, pp. 139-156, February 1992.
- [18] R. Ramakumar, *Engineering Reliability: Fundamentals and Applications*, Prentice-Hall, 1993.
- [19] K. R. Pattipati, Y. Li and H. A. P. Blom, "A unified framework for the performability of fault-tolerant computer systems." *IEEE Transactions on Computers*, vol. 42, pp. 312-326, March 1993.